



Automação de sistema robotizado colaborativo para soldadura de componentes para construção soldada

Luís Gonçalo Franco Ruas

Dissertação realizada no âmbito do
Mestrado Integrado em Engenharia Mecânica
Ramo de Automação

Orientador: Professor Doutor Germano Manuel Correia dos Santos Veiga
Coorientador: Professor Doutor António José Pessoa de Magalhães
Coorientador: Engenheiro João Ricardo Almeida de Sousa e Silva

setembro 2017

Resumo

A presente dissertação pretende projetar e implementar o sistema de segurança da célula robótica do projeto CoopWeld e desenvolver a automação não robótica da mesma, incluindo a solução de automação de um sistema hidráulico para movimentação e rotação de vigas.

O presente trabalho inicia-se pela definição da estratégia de segurança a adotar, fazendo a análise dos equipamentos da célula, normas aplicadas e conceitos base. A elaboração e constante atualização de um esquema da arquitetura da célula permitiu orientar e organizar todo o trabalho, servindo como elemento agregador de informação.

Os exercícios de avaliação e redução de riscos constituíram a base para a concepção de uma célula segura, onde os dispositivos de segurança foram programados para proteger o operador durante a fase de operação da máquina. Adicionalmente, foi idealizada e fundamentada a implementação de uma tarefa colaborativa entre o operador e o robô que permite ultrapassar um problema eminente do processo produtivo.

É realizado o estudo das características de funcionamento do mecanismo de movimentação e rotação de vigas. Seguindo-se o comissionamento de uma solução de sensorização para o mesmo.

Finalmente, é concebida a programação do equipamento utilizando o CodeSys para a programação do *soft*PLC, conseqüentemente é construída uma interface gráfica de controle manual implementada em C# e WPF de modo a integrar a aplicação de software da SARKKIS Robotics.

Automation of a collaborative robot system for welding of components for welded construction

Abstract

The present dissertation intends to design and implement the robotic cell safety system of the CoopWeld project and to develop the non-robotic automation of the cell, including the automation solution of a hydraulic system for handling and rotating beams.

The present work begins with the definition of the safety strategy to be adopted, analyzing the equipment of the cell, applied norms and basic concepts. The elaboration and the constant updating of the cell architecture schema allowed us to guide and organize all the work, serving as an aggregator of information.

The risk assessment and risk reduction exercises provided the basis for the design of a safe cell, where safety devices were programmed to protect the operator during the machine's operation phase. Additionally, the implementation of a collaborative task between the operator and the robot was idealized and well-founded to overcome an imminent problem of the production process.

It was carried out the study of the characteristics of operation of the movement and rotation of beams mechanism. Following the commencement of a sensor solution for the same.

Finally, using CodeSys for the programming of the softPLC, the equipment was programmed and, consequently a graphical interface of manual control was implemented in C# and WPF. This graphical interface was built to integrate the software application of SARKKIS Robotics.

Agradecimentos

Gostaria, em primeiro lugar, agradecer ao Professor Germano Veiga por me ter dado a oportunidade de desenvolver um tema que veio de encontro aos meus interesses, pelo constante apoio, disponibilidade e profissionalismo.

Um agradecimento à equipa SARKKIS Robotics, Engenheiros Pedro Malaca, Pedro Tavares, Daniel Marques, José Oliveira, João André Silva e, principalmente ao Engenheiro João Silva pelo exemplo, pelos esclarecimentos, disponibilidade e paciência demonstrados.

Dirigo ainda a minha gradidão aos amigos e colegas que me apoiaram durante estes 5 anos, em especial ao Jorge Cruz, Luís Costa e Luís Araújo.

À Catarina Pinto por ter estado sempre lá para mim.

E, principalmente, à minha família pelo apoio constante e por acreditarem. Sem eles nada seria possível.

Índice de Conteúdos

1	Introdução	1
1.1	Enquadramento e objetivos	1
1.2	Estrutura da dissertação	3
2	Estado da arte.....	5
2.1	Estratégia de segurança.....	5
2.1.1	Avaliação dos riscos – <i>Risk assessment</i>	7
2.1.2	Redução dos Riscos	15
2.1.3	Distâncias de segurança.....	17
2.2	Sistemas HRC	24
2.2.1	Requerimentos para operações colaborativas.....	25
2.3	Sistemas para rotação de vigas.....	26
2.3.1	FastRotator	26
2.3.2	Viradores de corrente.	27
2.3.3	Ocean Flipper	28
2.4	<i>SoftPL</i>	29
2.4.1	<i>SoftPLC</i> vs PLC convencional	30
2.4.2	Norma IEC 61131-3	31
2.5	Interface gráfica - WPF.....	38
3	Célula robótica – Arquitetura	41
3.1	Funcionamento previsto.....	41
3.2	Componentes e <i>Layout</i> da célula	41
3.3	Arquitetura do sistema	43
4	Segurança da célula robótica	45
4.1	Conceção da célula robótica	45
4.1.1	Zonas da célula	45
4.1.2	Procedimento operacional	46
4.1.3	Limites da máquina	47
4.1.4	Identificação dos perigos	49
4.1.5	Estimativa, Valoração e Redução dos riscos	51
4.1.6	Medidas de proteção adicionais OU Funções de controlo adicionais	68
4.1.7	Procedimento operacional – versão final.....	70
4.2	Tarefa colaborativa – <i>Hand guiding</i>	75
4.2.1	Exemplo demonstrativo do problema.....	75
4.2.2	Solução colaborativa	77
4.3	PLC de segurança.....	72
4.3.1	Escolha do equipamento.....	72
4.3.2	Programação e funcionalidades do programa.....	73
5	Projeto de automação dos Viradores	79
5.1	Virador – Machinery Services	79
5.1.1	Componente estrutural.....	80

5.1.2	Componente hidráulica.....	82
5.1.3	Princípio de funcionamento.....	85
5.1.4	Limitações	87
5.2	Controlo dos Viradores	87
5.2.1	Módulo de IO's.....	87
5.2.2	Teste do equipamento controlado pelo CodeSys.....	91
5.3	Sensorização.....	93
5.3.1	Tipos de sensores e detetores.....	93
5.3.2	Configuração	93
5.3.3	Solução escolhida e aplicação	96
5.3.4	Análise matemática	99
5.3.5	Validação com dispositivo real.....	106
5.4	Programação.....	109
5.4.1	Movimentos elementares – bloco de funções.....	109
5.4.2	Operações elementares – bloco de funções	110
5.4.3	Programa <i>Main</i>	111
5.4.4	Modo manual.....	113
5.4.5	Modo automático – bloco de funções.....	113
5.4.6	Emergência	116
6	Interface Gráfica - <i>SoftPLC</i>	117
7	Conclusões e Trabalhos Futuros	121
7.1	Trabalhos Futuros	122
8	Referências.....	123
ANEXO A	<i>Template</i> - Avaliação e Redução de riscos	127
ANEXO B	Projeto dos suportes para os lasers	126
ANEXO C	Programa de controlo dos viradores	128

Índice de Figuras

Figura 1 - Célula robótica: 1-Elementos de segurança; 2-HMI e bancada móvel; 3-Viradores; 4-Robô e semipórtico; 5-Eixo linear (Track); 6-Controlador robótico e quadros elétricos. ----	2
Figura 2 - Esquema representativo da estratégia de segurança segundo a norma ISO 12100-1.6	
Figura 3 - Eventos perigosos e consequências da exposição prolongada ao perigo. -----	9
Figura 4 - Representação gráfica das abordagens top-down e bottom-up. -----	10
Figura 5 - IEC 62061: Componentes Se e Probabilidade de ocorrência. -----	11
Figura 6 - EN ISO 13849-1: Esquema da atribuição do PL. -----	14
Figura 7 - Distância de segurança: zona de detecção ortogonal à direção de aproximação. ---	18
Figura 8 - Distância de segurança: zona de detecção paralela à direção de aproximação. ----	19
Figura 9 - Distância de segurança: zona de detecção inclinada relativamente à direção de aproximação. -----	20
Figura 10 - Distância de segurança: alcance através do contorno do equipamento eletrosensitivo. -----	21
Figura 11 - Distância de segurança: alcance através do contorno de estruturas de proteção.-	22
Figura 12 - Exemplo de colaboração entre o operador e o robô. -----	24
Figura 13 - Exemplo de cooperação entre o operador e o robô. -----	24
Figura 14 - Exemplo de coexistência entre o operador e o robô. -----	25
Figura 15 - FastRotator FR2012: capacidade 12 ton., rotação 90° em 30s, diâmetro de viragem máximo de 2800mm. -----	26
Figura 16 - FastRotator FR1508: capacidade 8 ton., rotação 180° em 18s, diâmetro de viragem máximo de 2100mm. -----	26
Figura 17 – Virador de corrente: Stierli-Bieger Rotator 1000, cap. 12 ton., rotação 90° em 15s, diâmetro de viragem máximo de 1000mm. -----	27
Figura 18 – Ocean Flipper: capacidade 12 ton., rotação 90° em 15s, dimensão máxima da viga de 1000mm. -----	28
Figura 19 - Hierarquias de chamada das POU. -----	33
Figura 20 - Modelo de software do PLC de acordo com a norma IEC 61131. -----	33
Figura 21 - Exemplo de representações gráficas de declarações de tarefas. À esquerda é representado um bloco genérico, ao centro um bloco cíclico e à direita um bloco não cíclico. Esta representação gráfica não é contemplada pela norma IEC 61131. -----	34
Figura 22 - Exemplo de instruções em IL. -----	35
Figura 23 - Exemplo de instruções em ST.-----	35
Figura 24 - Exemplo de elementos de um FBD.-----	36
Figura 25 - Exemplo de elementos de um LD.-----	36

Figura 26 - Exemplo de um programa em SFC. -----	37
Figura 27 - WPF: Alteração da aparência dos botões. -----	38
Figura 28 - WPF: Exemplo onde a aparência é dependente da resolução. -----	38
Figura 29 - Célula Robótica: Layout inicial. -----	41
Figura 30 - Célula robótica: Fluxo de informação e Arquitetura do sistema. -----	43
Figura 31 - Zonas da célula robótica: Vermelho-Zona de trabalho do robô; Verde-Zona de trabalho do operador; Azul-Vias de circulação. -----	45
Figura 32 - Identificação das barreiras de segurança (vista de topo). -----	53
Figura 33 - Pontos potencialmente perigosos da Barreira Esquerda. -----	54
Figura 34 - Distância mínima de segurança para a Barreira Esquerda. -----	55
Figura 35 - Pontos potencialmente perigosos da Barreira Frente. -----	55
Figura 36 - Distância mínima de segurança para a Barreira Frente. -----	56
Figura 37 - Pontos potencialmente perigosos da Barreira Direita. -----	56
Figura 38 - Distância mínima de segurança para a Barreira Direita. -----	57
Figura 39 - Configuração nº1: Distância mínima de segurança para a Barreira Luminosa.---	59
Figura 40 - Barreira luminosa ReeR EOS4 A 40 mm resolution model 1354. -----	59
Figura 41 - Configuração nº1: Limites da área de trabalho segundo o eixo X considerando o pórtico posicionado o mais próximo da Barreira Esquerda.-----	60
Figura 42 – Configuração nº1: limites da área de trabalho segundo o eixo Y considerando o pórtico posicionado o mais próximo da Barreira Direita. -----	61
Figura 43 – Configuração nº1: Análise do espaço disponível para a bancada, vista de topo. -----	62
Figura 44 – Configuração nº1: Carro de transporte KARIZER+KRAFT. -----	64
Figura 45 – Configuração nº1: Conjunto de roda para carril Blickle.-----	65
Figura 46 - Estrutura de suporte à HMI. Componentes Rittal.-----	67
Figura 47 - Componentes da estrutura Rittal da HMI. -----	68
Figura 48 - Dispositivos de proteção da HMI: 1-Botão Enable; 2-Botoneira de emergência; 3-Botão de rearme. -----	69
Figura 49 - Célula robótica parcial: 1-Track; 2-Cavalete auxiliar; 3-Componentes a soldar; 4-Semipórtico; 5-Robô; 6-Projetor laser. -----	75
Figura 50 - Célula robótica parcial: Componentes a soldar. -----	76
Figura 51 - Célula robótica parcial: Projeção do perfil nº2 e da chapa nº3.-----	76
Figura 52 - Célula robótica parcial: Projeção falhada da chapa nº3. -----	76
Figura 53 - Programa do PLC de Segurança. -----	74
Figura 54 - Fotografia dos Viradores durante a fase de montagem da célula. -----	79
Figura 55 - Representação CAD do modelo do Virador. -----	80
Figura 56 - Representação CAD detalhada do modelo do Virador.-----	80
Figura 57 - Representação CAD da Carruagem interior. -----	81
Figura 58 - Representação CAD dos detalhes interiores da carcaça exterior. -----	82

Figura 59 - Esquema hidráulico simplificado do Virador.	83
Figura 60 – Fotografia da central hidráulica do Virador.	84
Figura 61 – Rotação Positiva Etapa 0.	85
Figura 62 - Rotação Positiva Etapa 1.	85
Figura 63 - Rotação Positiva Etapa 2.	85
Figura 64 - Rotação Positiva Etapa 3.	85
Figura 65 - Rotação Negativa Etapa 0.	86
Figura 66 - Rotação Negativa Etapa 1.	86
Figura 67 - Mov. Horizontal Etapa 2.	86
Figura 68 - Mov. Horizontal Etapa 3.	87
Figura 69 - Módulos WAGO.	88
Figura 70 – CodeSys: Adaptador Ethernet.	89
Figura 71 - Codesys: Árvore de dispositivos.	90
Figura 72 - CodeSys: Configuração adapter genérico.	91
Figura 73 - Sensores - configuração nº 1.	94
Figura 74 - Sensores - configuração nº 2.	94
Figura 75 - Sensores - configuração nº 3.	95
Figura 76 - Sensores - configuração nº 4.	95
Figura 77- Posicionamento dos sensores, vista 2D.	98
Figura 78 - Posicionamento dos sensores, vista 3D.	98
Figura 79 - Laser 1 - Relações trigonométricas.	99
Figura 80 - Laser 1 - Triângulo DGH.	101
Figura 81 - Laser 1 - Gráfico do valor do sensor em função do ângulo da pá para várias inclinações do Laser 1.	102
Figura 82 - Laser 1 - Aproximação polinomial do ângulo da pá em função do valor do sensor recebido.	103
Figura 83 - Laser 2 - Relações trigonométricas.	103
Figura 84 - Laser 2 - Aproximação polinomial do ângulo da pá em função do valor do sensor recebido.	105
Figura 85 - Programação Viradores: Bloco de funções dos movimentos elementares.	109
Figura 86 - Programação Viradores: G21 – Movimentos elementares.	110
Figura 87 - Programação Viradores: Bloco de funções das operações elementares.	110
Figura 88 - Programação Viradores: G22 – Operações elementares.	111
Figura 89 - Programação Viradores: Macro etapa M10 do G22.	111
Figura 90 - Programação Viradores: G1 – Programa Main.	112
Figura 91 - Programação Viradores: G2 – Modo Manual.	113
Figura 92 - Programação Viradores: Bloco de funções do modo automático.	114
Figura 93 - Programação Viradores: G3 – Modo Automático.	114

Figura 94 - Programação Viradores: Macro etapa M10 do G3. -----	114
Figura 95 - Interface gráfica: Vizualisation do CodeSys. -----	117
Figura 96 - Interface gráfica: WPF. -----	119

Índice de Tabelas

Tabela 1 - IEC 62061 e EN ISO 13849-1: Aplicações.....	11
Tabela 2 - IEC 62061: Severidade da potencial consequência.....	12
Tabela 3 - IEC 62061: Frequência e tempo de exposição ao perigo	12
Tabela 4 - IEC 62061: Probabilidade de ocorrência de um evento ou situação perigosa	12
Tabela 5 - IEC 62061: Probabilidade de evitar ou limitar o perigo	13
Tabela 6 - IEC 62061: Classificação SIL	13
Tabela 7 - IEC 62061: Probabilidade de ocorrência de falha por hora	13
Tabela 8 - Eficácia das várias medidas de proteção	15
Tabela 9 - Distância de segurança: distância adicional para a situação do alcance através do contorno do equipamento eletrosensitivo	21
Tabela 10 - Distância de segurança: distância para a situação do alcance através do contorno de uma estrutura de proteção para riscos reduzidos	23
Tabela 11 - Distância de segurança: Distância para a situação do alcance através do contorno de uma estrutura de proteção para riscos elevados.....	23
Tabela 12 Célula robótica: Componentes base.....	42
Tabela 13 - Identificação dos perigos.....	49
Tabela 14 - Estimativa, Valoração e Redução dos riscos.....	51
Tabela 15 - Distâncias mínimas de segurança para as Barreira Esquerda, Barreira Direita e Barreira Frente.....	54
Tabela 16 - Configuração nº1: Valor de <i>CRO</i> para a Barreira Luminosa	58
Tabela 17 – Configuração nº1: Características carro de transporte KARIZER+KRAFT	64
Tabela 18 – Configuração nº1: Comparação entre o material do piso para rodas de rodízios fixos	64
Tabela 19 – Configuração nº1: Rodas para diferentes tipos de carris	65
Tabela 20 – Configuração nº1: Componentes da estrutura Rittal da HMI.....	68
Tabela 21 – Configuração nº1: dispositivos de segurança adicionais	69
Tabela 22 – Lista dos componentes hidráulicos do Virador	83
Tabela 23 - Lista de módulos WAGO necessários.....	88
Tabela 24 - Tempos de subida e descida da Pá esquerda e da Pá direita	92
Tabela 25 - Comparação entre os sensores considerados.....	96
Tabela 26 - Sensores Sick DT20 HI e SICK OD2	97
Tabela 27 – Comparação dos ângulos teóricos e reais das pás do Virador	107

Tabela 28 - Comparação da posição teórica e real da posição da carruagem	108
Tabela 30 - Projeto dos suportes dos Lasers	134

1 Introdução

Com o avanço da tecnologia, a utilização de robôs e a automatização de processos tem substituído em grande parte a presença humana nas linhas de produção. A aplicação dos robôs na soldadura é, de longe, a aplicação mundialmente mais utilizada, principalmente na indústria automóvel.

Hoje em dia, os robôs são componentes essenciais de uma fábrica, e cada vez mais importantes no caminho para as Fábricas do Futuro.

Um novo tipo de robô tem vindo a ser desenvolvido, juntamente com um novo conceito, os robôs colaborativos. Pretendem ser mais inteligentes, mais flexíveis, seguros e fáceis de utilizar [1] [2] [3].

1.1 Enquadramento e objetivos

Esta dissertação está enquadrada no projeto CoopWeld (projeto P2020) que está a ser desenvolvido por um consórcio liderado pela SARKKIS Robotics conjuntamente com o INESC-TEC, ISQ e NORFER.

Este projeto consiste no desenvolvimento de um sistema robótico para soldadura colaborativa de componentes em vigas de construção soldada com integração de tecnologia BIM e realidade aumentada. No final do projeto, a célula robótica deverá estar finalizada e instalada nas instalações da NORFER Construções.

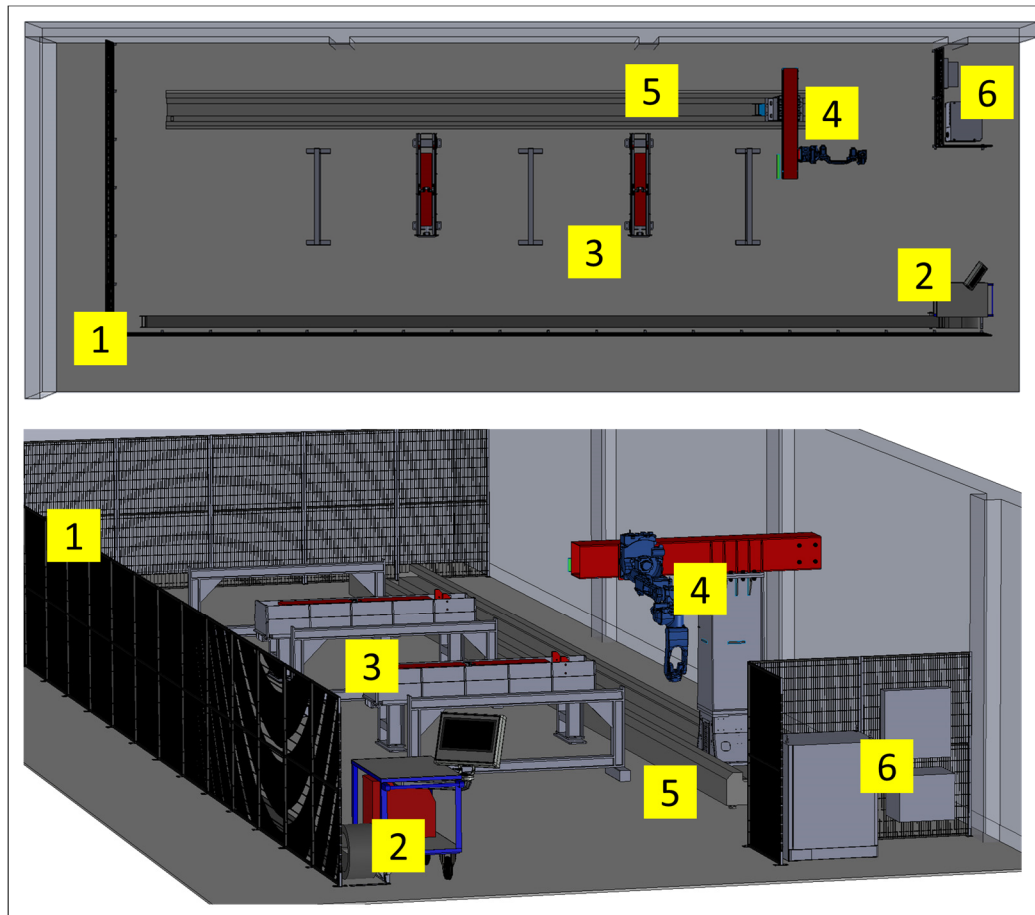


Figura 1 - Célula robótica: 1-Elementos de segurança; 2-HMI e bancada móvel; 3-Viradores; 4-Robô e semipórtico; 5-Eixo linear (*Track*); 6-Controlador robótico e quadros elétricos.

A Figura 1 apresenta a célula robótica, em fase terminal, e destaca alguns componentes relevantes sobre os quais é desenvolvida esta dissertação.

Nesta dissertação, pretende-se projetar e implementar uma solução para o sistema de segurança (nº1 da Figura 1) da célula robótica e desenvolver a automação não robótica da célula incluindo a solução de automação para os Viradores (nº3 da Figura 1), sendo estes responsáveis pela movimentação e rotação das vigas a serem soldadas.

Objetivos

- Desenvolvimento da lógica do equipamento e orquestração com controlador robótico para a operação do sistema.
- Automação dos elementos não robóticos da célula industrial.
- Avaliação de requisitos de segurança e documentação.
- Conceção, desenvolvimento e coordenação de fabrico de carrinho de transporte de componentes com interface gráfica incorporada para operação do sistema colaborativo.
- Desenvolvimento da interface de operação manual do equipamento.

1.2 Estrutura da dissertação

Esta dissertação encontra-se dividida em 6 capítulos.

O capítulo 1 faz uma breve apresentação da célula em estudo descrevendo as áreas de incidência deste trabalho.

No capítulo 2 é apresentado o estado da arte utilizado.

O capítulo 3 descreve os componentes principais e apresenta um esquema que agrega todos os fluxos de informação e componentes da totalidade da célula robótica.

No capítulo 4 é descrita a estratégia de segurança adotada e que serve de base para a conceção da célula.

Nos capítulos 5 e 6 é feita a automação dos Viradores, desde o estudo do equipamento à criação da interface gráfica que o controla.

2 Estado da arte

No presente estado pretende-se aprofundar o conhecimento acerca de todos os elementos que são importantes para esta tese. Assim, e recorrendo à Figura 1 tratar-se-á de: estudar, do ponto de vista da segurança e seguindo as respetivas normas, como é concebida a célula (nº1 da Figura 1) e como é que esta protege o Humano dos perigos nela presentes (nºs 3, 4 e 5 da Figura 1); conhecer os diferentes sistemas de rotação e movimentação de vigas (nº3 da Figura 1), realizando uma análise comparativa entre eles, e ainda perceber quais são as possibilidades de programação (nºs 2 e 6 da Figura 1) do equipamento escolhido que podem ser aplicadas na célula em estudo.

2.1 Estratégia de segurança

A norma ISO 12100-1 [4] apresenta uma metodologia para o projeto de uma solução de segurança para a generalidade das máquinas. É dividida em duas partes principais, a avaliação dos riscos, ou *risk assessment*, e a redução dos riscos. Esta é apresentada, graficamente, pela Figura 2 [4] [5].

Em primeiro lugar deve ser realizado o exercício da avaliação dos riscos, começando com a especificação dos limites de uso e com os limites espaciais e temporais da máquina, seguindo-se a identificação dos perigos existentes.

De seguida, é feita a estimativa dos riscos inerentes a cada perigo identificado, quantificando-o segundo uma ou mais metodologias de estimação de risco e, por fim, é feita a valoração do risco, que dita a necessidade de medidas de redução do risco ou o término de todo o processo.

A redução do risco é feita seguindo uma metodologia de três etapas. A primeira etapa consiste na remoção, se possível, do risco por modificações na conceção da máquina. Todavia, na impossibilidade da sua eliminação é feita a sua redução, recorrendo a modificações na conceção da máquina.

Inicia-se a segunda etapa caso os riscos residuais da primeira etapa sejam demasiado acentuados. Assim, a redução do risco passa pela implementação de medidas de proteção adequadas.

Caso, após a segunda etapa, existam, ainda, riscos residuais, estes devem constituir nas informações de utilização da máquina, sendo esta a terceira etapa do processo de redução de risco.

Todo este processo é iterativo, ou seja, sempre que é tomada uma medida de redução de risco, o exercício de avaliação dos riscos deve ser refeito para avaliar se o risco passou a ser tolerável e/ou se foram gerados novos riscos [4].

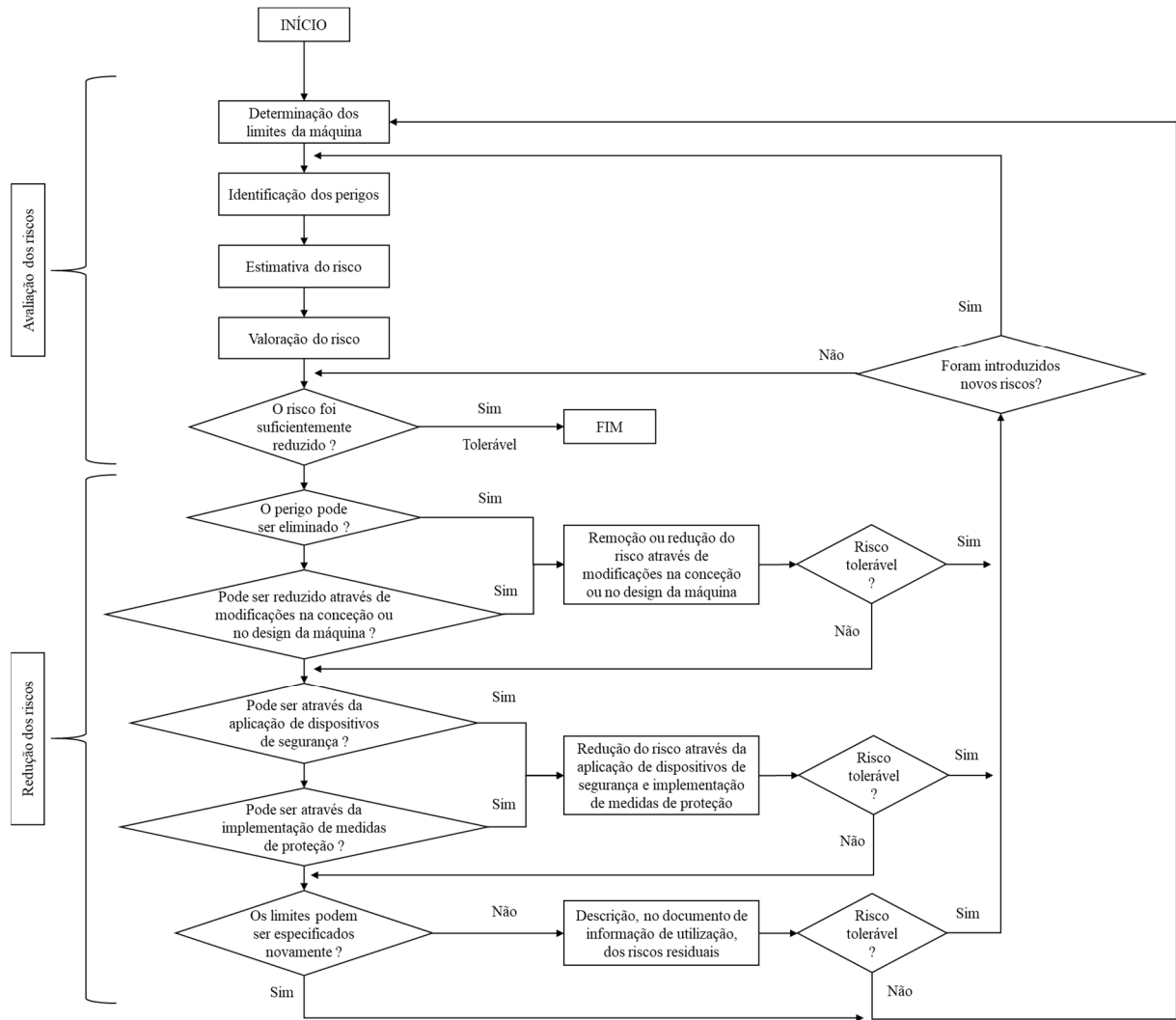


Figura 2 - Esquema representativo da estratégia de segurança segundo a norma ISO 12100-1, [4].

2.1.1 Avaliação dos riscos - *Risk assessment*

A avaliação dos riscos, tal como já foi referido, é um processo iterativo e deve ser feito nas várias fases do ciclo de vida de uma máquina, desde a conceção ao desmantelamento.

Para uma boa avaliação deve ter-se acesso a:

- Especificações do utilizador
- Informação sobre a máquina:
 - Várias fases do ciclo de vida
 - Desenho dos equipamentos
 - Fontes de energia utilizadas
 - Documentação sobre outras máquinas similares
 - Informações de utilização (caso existam)
- Normas, diretivas e outros documentos técnicos relevantes
- Relatos de experiências anteriores
- Princípios de ergonomia

A informação deve ser atualizada sempre que se apliquem modificações ou novos desenvolvimentos [3].

Especificação dos limites

Deve ser considerada toda a informação acerca das peças constituintes, dos mecanismos e funções da máquina, de todas as tarefas que envolvam interação com o humano e com o ambiente onde será instalada a máquina.

A combinação de duas ou mais máquinas constitui uma só máquina e deve ser vista como um todo.

Todas as condições normais de utilização e má utilização razoavelmente previsível devem ser consideradas, assim como o mal funcionamento da máquina.

De seguida, são apresentados todos os aspetos a ter em conta na especificação dos limites da máquina [6] [7]:

Limites de utilização

- Descrição de funções, utilização prevista e má utilização razoavelmente previsível.
- Diferentes modos de operação, incluindo intervenções devido à manutenção requerida ou a avarias.
- Descrição dos equipamentos e ferramentas.
- Quais as fontes de energia presentes.
- Descrição das características requeridas para o operador ou para qualquer outro interveniente durante o ciclo de vida da máquina, tais como, utilizadores alvo, nível de treino requerido, experiência, entre outras.
- Extensão do perigo a outras pessoas na vizinhança da máquina.

Limites espaciais

- Área de movimentação.

- Espaço requerido para o operador e outros intervenientes durante as etapas de operação e manutenção.
- Interfaces de interação com o humano como, por exemplo, HMI's ou quadros elétricos.

Limites temporais

- Tempo de duração de ferramentas ou peças de desgaste.
- Tempos de ciclo do processo produtivo.
- Intervalos de manutenção.

Outros Limites

- Condições ambientais.
- Ambiente de funcionamento.
- Material a ser processado.
- Outras experiências anteriores.

Identificação dos perigos

O objetivo desta etapa é a constituição de uma lista de perigos, situações perigosas e eventos perigosos e possíveis consequências.

Esta etapa pode ser considerada como a mais crítica em todo o processo, na medida em que, um perigo não identificado é um perigo não avaliado e, conseqüentemente, não controlado.

Esta é a etapa mais importante da avaliação de riscos, pois só quando um perigo é identificado é que se podem tomar medidas de redução do risco. Portanto, esta etapa deve ser sistematizada e intensiva.

Deve contemplar diferentes cenários que possam ser perigosos para o homem qualquer seja a sua natureza, tendo em conta o local de trabalho, operações rotineiras e não rotineiras, eventos não planeados, mas previsíveis, ou seja, ter em conta todas as fases e tarefas que possam acontecer durante todo o ciclo de vida da máquina.

A Figura 3 demonstra em que contexto é que eventos perigosos podem ser gerados e quando é que o humano pode sofrer consequências de um perigo.

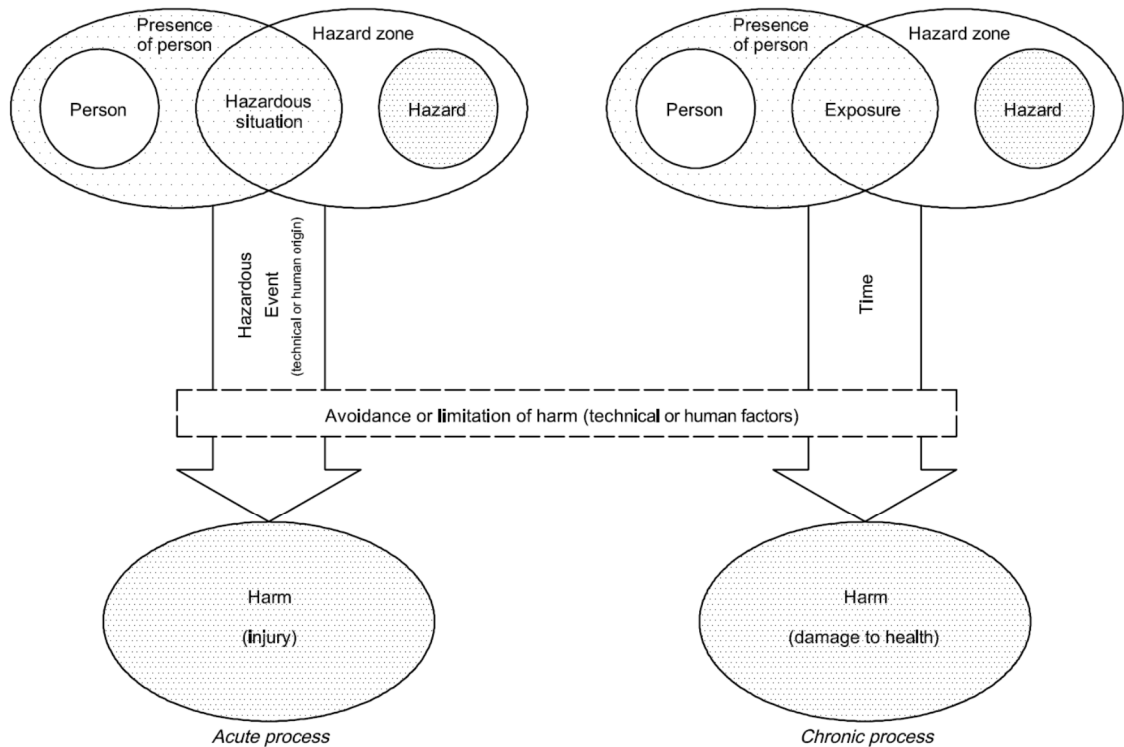


Figura 3 - Eventos perigosos e consequências da exposição prolongada ao perigo [8].

A zona perigosa é a zona onde se pode encontrar um determinado perigo e, quando esta é coincidente com a zona de movimento de uma pessoa gera-se uma situação perigosa. Esta situação perigosa pode, eventualmente, dar origem a um evento perigoso como, por exemplo, uma fuga de óleo ou um curto circuito que possam ferir ou causar algum dano ao humano.

O tempo de exposição ao perigo é, também, um fator de impacto na saúde do Homem. Por exemplo, quando uma pessoa se encontra numa zona perigosa exposta à radiação ou à luz intensa, é uma questão de tempo até sofrer consequências na sua saúde, normalmente problemas associados à cronicidade.

Assim, esta etapa deve ser convenientemente planeada e organizada. Existem vários métodos recursivos que, em geral, seguem uma abordagem *top-down* ou *bottom-up*. A Figura 4 apresenta, graficamente, os elementos que se devem considerar e demonstra o funcionamento destas duas abordagens.

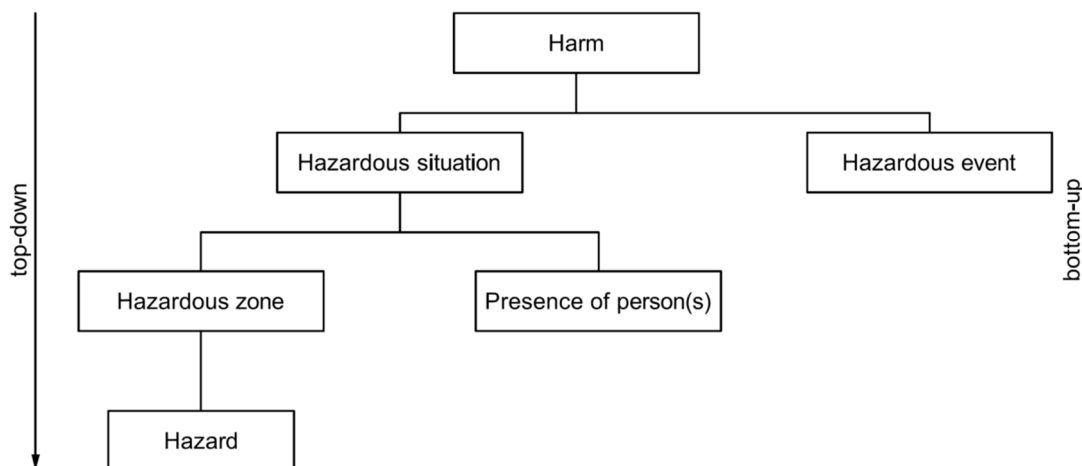


Figura 4 - Representação gráfica das abordagens *top-down* e *bottom-up* [8].

A abordagem *top-down* inicializa-se nas possíveis consequências (corte, esmagamento, entre outros) e procura saber qual a causa, ou seja, identifica o evento perigoso, a situação perigosa e, por fim, identifica onde está e qual é o perigo que está na origem.

A abordagem *bottom-up*, por outro lado, inicia-se pela análise de todos os perigos, considerando todas as situações perigosas que possam acontecer, chegando, assim, às consequências dos potenciais perigos. Esta abordagem é mais lógica, rápida e fácil de concretizar por indivíduos menos experientes [5] [6] [8].

Estimativa do Risco

Nesta etapa, pretende-se quantificar o risco, classificando-o com a atribuição de um nível ou índice.

Quanto maior for a potencialidade de um evento perigoso ou situação perigosa, mais importante é reduzir o risco respetivo [5] [8].

Existem vários métodos normalizados para realizar esta quantificação, um é proposto pela norma IEC 62061 (*Safety Integrity Levels – SIL*) e outro pela EN ISO 138449-1 (*Performance Levels – PL*) [9].

Ambas aplicam o conceito *functional safety*, que visa simplificar os requisitos, ou seja, abordar o perigo e as situações, ou eventos, de maneira simples e direta, como, por exemplo, se uma barreira luminosa for cruzada, o movimento perigoso deverá parar.

A principal diferença entre estas duas metodologias encontra-se no campo de aplicação alvo, pois cada uma diz respeito ao contexto da norma que lhe deu origem. Porém, ambas as metodologias produzem resultados mais ao menos iguais, quando empregadas em qualquer aplicação [10].

Tabela 1 - IEC 62061 e EN ISO 13849-1: Aplicações [10]

	Technology implementing the safety-related control function(s)	ISO 13849-1 (under revision)	IEC 62061
A	Non electrical, e.g. hydraulics	X	Not covered
B	Electromechanical, e.g. relays, or non complex electronics	Restricted to designated architectures (see Note 1) and up to PL=e	All architectures and up to SIL 3
C	Complex electronics, e.g. programmable	Restricted to designated architectures (see Note 1) and up to PL=d	All architectures and up to SIL 3
D	A combined with B	Restricted to designated architectures (see Note 1) and up to PL=e	X see Note 3
E	C combined with B	Restricted to designated architectures (see Note 1) and up to PL=d	All architectures and up to SIL 3
F	C combined with A, or C combined with A and B	X see Note 2	X see Note 3

"X" indicates that this item is dealt with by the standard shown in the column heading.

NOTE 1 Designated architectures are defined in Annex B of EN ISO 13849-1(rev.) to give a simplified approach for quantification of performance level.

NOTE 2 For complex electronics: Use of designated architectures according to EN ISO 13849-1(rev.) up to PL=d or any architecture according to IEC 62061.

NOTE 3 For non-electrical technology use parts according to EN ISO 13849-1(rev.) as subsystems.

Tal como pode ser interpretado pela Tabela 1, a norma IEC 62061 não foi concebida para tratar aplicações não elétricas, mas a ISO13849-1 já engloba esse aspeto.

IEC 62061 - SIL

A quantificação SIL quantifica o perigo recorrendo a dois identificadores: a severidade da potencial consequência (*Se*) e a probabilidade de ocorrência do evento ou situação perigosa. A Figura 5 exemplifica esta abordagem [9].

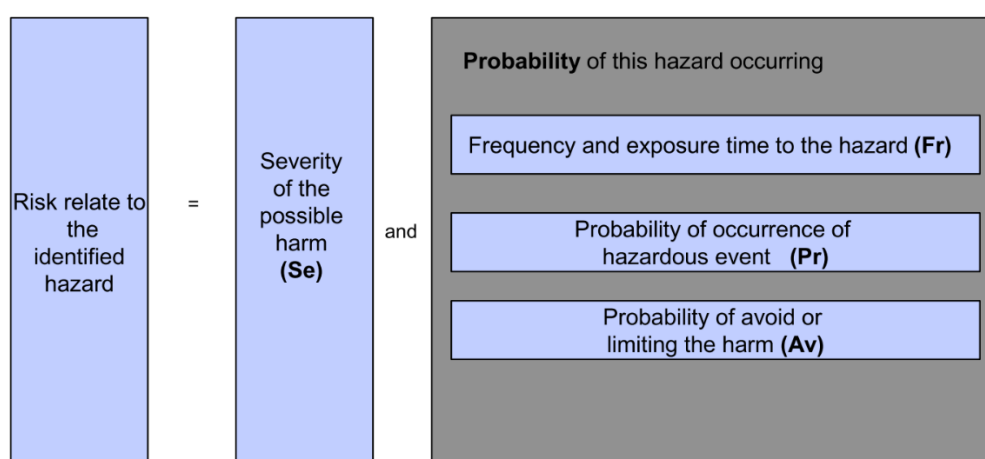


Figura 5 - IEC 62061: Componentes Se e Probabilidade de ocorrência [9].

A severidade da potencial consequência é quantificada segundo 4 níveis, como é possível verificar na Tabela 2 - IEC 62061: Severidade da potencial consequência Tabela 2.

Tabela 2 - IEC 62061: Severidade da potencial consequência [10]

Consequências	Severidade – <i>Se</i>
Irreversível: morte, perda de um olho ou braço	4
Irreversível: perna partida, perda de um dedo	3
Reversível: requer atendimento médico	2
Reversível: requer primeiros socorros	1

A probabilidade de ocorrência do evento ou situação perigosa é classificada em diferentes classes, classes *CI*, e corresponde à soma de três parâmetros.

$$CI = Fr + Pr + Av \quad (1)$$

Fr – Frequência e tempo de exposição ao perigo

Pr – Probabilidade de ocorrência de um evento ou situação perigosa

Av – Probabilidade de evitar ou limitar o perigo

As tabelas seguintes permitem quantificar estes três parâmetros. A Tabela 3 refere-se à frequência e tempo de exposição ao perigo, a Tabela 4 refere-se à probabilidade de ocorrência de um evento ou situação perigosa e a Tabela 5 refere-se à probabilidade de evitar ou limitar o perigo.

Tabela 3 - IEC 62061: Frequência e tempo de exposição ao perigo [7]

Periodicidade da exposição	Duração	
	< 10 min	> 10 min
$\leq 1h$	$Fr = 5$	$Fr = 5$
$> 1h$ to ≤ 1 dia	$Fr = 4$	$Fr = 5$
> 1 dia to ≤ 2 semanas	$Fr = 3$	$Fr = 4$
> 2 semanas to ≤ 1 ano	$Fr = 2$	$Fr = 3$
> 1 ano	$Fr = 1$	$Fr = 2$

Tabela 4 - IEC 62061: Probabilidade de ocorrência de um evento ou situação perigosa [7]

Probabilidade de ocorrência	<i>Pr</i>
Muito provável	5
Provável	4
Possível	3
Raramente	2
Desprezável	1

Tabela 5 - IEC 62061: Probabilidade de evitar ou limitar o perigo [7]

Probabilidade de evitar ou limitar	<i>Av</i>
Impossível	5
Raramente	3
Provável	1

O valor SIL do perigo analisado é, então, atingido através do cruzamento entre os valores de *Se* e *CI* da Tabela 6.

Tabela 6 - IEC 62061: Classificação SIL [7]

<i>Se</i>	Classe <i>CI</i>				
	3 – 4	5 – 7	8 – 10	11 – 13	14 – 15
4	SIL 2	SIL 2	SIL 2	SIL 3	SIL 3
3	-	OM	SIL 1	SIL 2	SIL 3
2	-	-	OM	SIL 1	SIL 2
1	-	-	-	OM	SIL 1

As áreas a cinza claro, OM de outras medidas, não dispensam a redução do risco, mas também não impõe uma classificação fixa, apenas assinalam a utilização de outras medidas de redução.

A classificação SIL pode ser traduzida na probabilidade de falha por hora, recorrendo à seguinte tabela, Tabela 7, onde PFHD significa *Probability of a Dangerous Failure per Hour* [10].

Tabela 7 - IEC 62061: Probabilidade de ocorrência de falha perigosa por hora [7]

SIL	<i>PFHD</i>
1	$\geq 10^{-6} \text{ to } < 10^{-5}$
2	$\geq 10^{-7} \text{ to } < 10^{-6}$
3	$\geq 10^{-8} \text{ to } < 10^{-7}$

EN ISO 138449-1 - PL

Comparativamente com a anterior, esta metodologia é apresentada de forma mais simples. A Figura 6 resume integralmente esta metodologia.

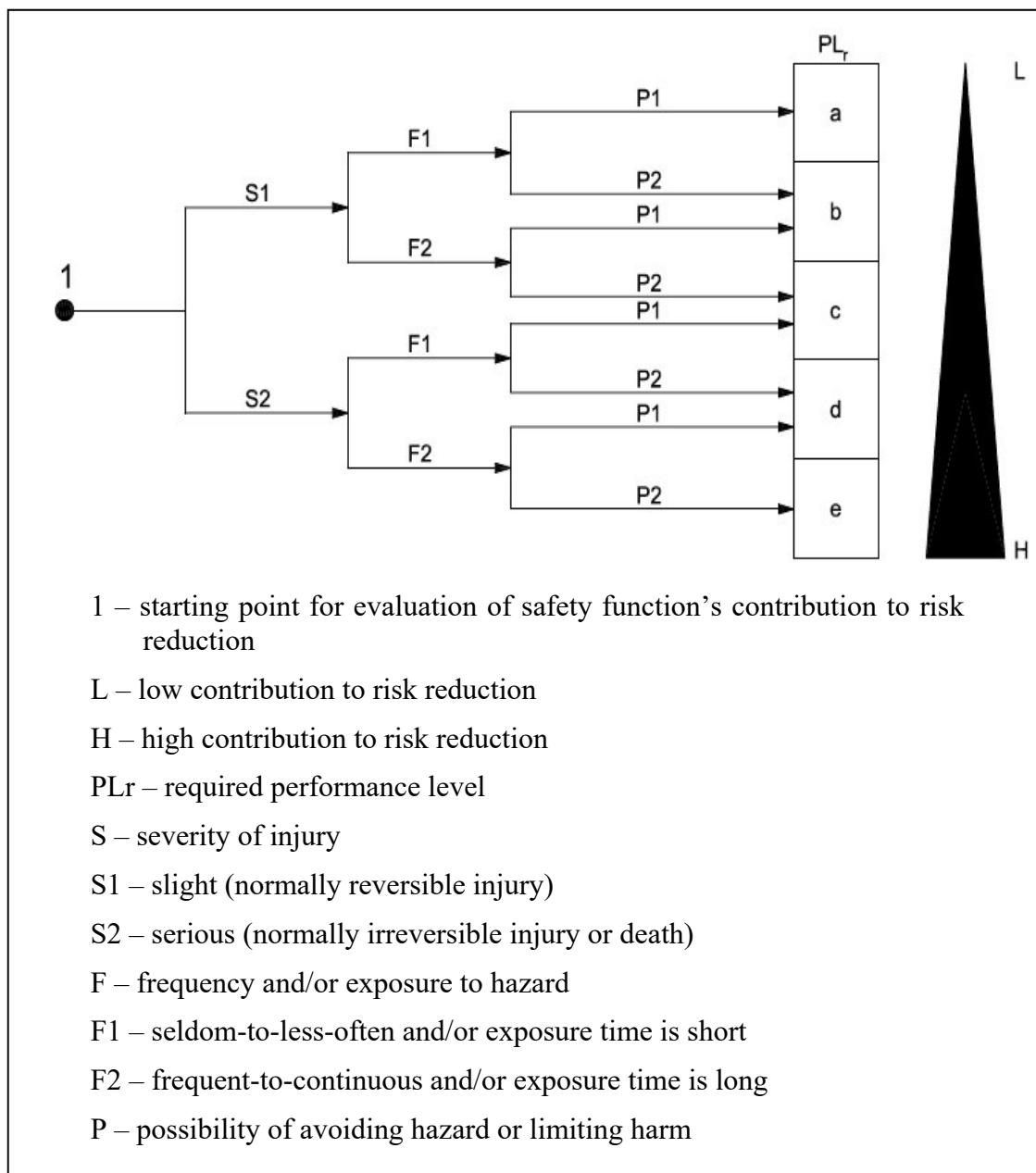


Figura 6 - EN ISO 13849-1: Esquema da atribuição do PL [11].

A quantificação é conseguida segundo um caminho de escolhas. A primeira escolha consiste em avaliar a severidade do perigo: S1 se o perigo for pouco severo ou reversível ou S2 se o perigo for de maior severidade, correspondendo a consequências irreversíveis ou até a morte. A segunda etapa é corresponde à frequência de exposição ao perigo: F1 para perigos pouco frequentes ou F2 para perigos contínuos ou frequentes. A terceira e última etapa de escolhas consiste em determinar se o perigo pode ser evitado ou as consequências podem ser limitadas: P1 quando existe essa possibilidade sobre determinadas condições ou P2 quando é dificilmente possível.

Valoração do risco

O objetivo desta etapa é decidir se o risco que está a ser analisado é ou não tolerável. Caso não seja tolerável, parte-se para a etapa de redução dos riscos [8].

2.1.2 Redução dos Riscos

Tal como referido anteriormente e representado na Figura 2, a norma 12100-1 propõe um método de redução dos riscos em três etapas [6]:

1. Remoção do perigo ou redução do risco através de modificações na conceção da máquina
2. Aplicação de dispositivos de segurança e implementação de medidas de proteção
3. Descrição, no documento de informação de utilização, dos riscos residuais das etapas anteriores

A norma ISO/TR 14121-2 apresenta, na Tabela 8, a eficácia das várias medidas na redução de risco, iniciando-se na mais eficaz, prioridade 1, até a menos eficaz, prioridade 4 [8].

Tabela 8 - Eficácia das várias medidas de proteção [8]

Medida	Prioridade	Alternativa
Eliminação do risco	1	Redução da severidade das possíveis consequências
Eliminação da situação perigosa	2	Redução da frequência ou duração da exposição ao perigo
Eliminação dos possíveis eventos perigosos	3	Redução da probabilidade de ocorrência de eventuais eventos perigosos
Implementação de medidas para evitar os perigos	4	Implementação de medidas para limitar o perigo

Remoção do perigo por modificações na conceção da máquina

O primeiro passo passa pela eliminação total do perigo. Para tal, podem, por exemplo, tomar-se as seguintes ações [6] [12]:

- Substituição de materiais ou substâncias perigosas
- Modificações físicas (arestas afiadas, elementos pontegudos)
- Eliminação ou modificação da atividade respetiva
- Modificações de *layout*
- Automação de tarefas

Redução do perigo por modificações na conceção da máquina

Não sendo possível a remoção integral do perigo, podem utilizar-se medidas de redução do perigo através de modificações na conceção da máquina, como por exemplo [6] [12]:

- redução dos níveis de energia, ou seja, forças menores, velocidades mais reduzidas, pressões mais baixas, etc.
- recolocação das fontes de perigo
- utilização de componentes mais fiáveis e seguros
- aplicar medidas de proteção aos sistemas de controlo dos equipamentos (redundância, por exemplo)

Aplicação de dispositivos de segurança

Se, através das modificações na conceção da máquina, os riscos continuarem intoleráveis, deve aplicar-se dispositivos de segurança para concretizar a redução dos riscos. Alguns dos dispositivos mais comumente utilizados são [6] [12]:

- Barreiras físicas, gradeamento e jaulas
- Protetores móveis com encravamento (*interlocking guards*)
- Equipamento de proteção eletrosensível (barreiras luminosas, *scanners* de área, tapetas sensíveis à pressão)
- Dispositivos de segurança relacionados com o sistema de controlo (*enabling device*, controlo de movimento limitado, botoneiras *press-to-run*)
- Dispositivos de limitação (limitação de correntes, pressões, velocidade, etc.)

Implementação de medidas de proteção

Não se tendo obtido níveis de risco toleráveis com a aplicação dos dispositivos e modificações apresentados anteriormente, resta implementar medidas de proteção, como, por exemplo [6] [12]:

- Botoneiras de emergência
- Sinais luminosos e sonoros
- Medidas para entrar e sair da célula em situações de emergência
- Medidas de controlo de acesso à célula
- Meios para facilitar o manuseamento dos equipamentos e materiais

Informações de utilização

Por último, todos os riscos residuais das etapas anteriores devem ser informados ao utilizador. Existem, assim, informações na própria máquina e informações que devem acompanhar a máquina [6] [12].

➤ Informações na máquina [6] [12]

- Sinais sinalizadores de perigo
- Sinalização sonora e auditiva
- Marcações e legendas para um uso seguro

➤ Informações que acompanham a máquina [6] [12]

- Manual de instruções
- *Datasheets*
- Informação acerca da necessidade de treino dos operadores
- Equipamento de proteção pessoal adequado
- Procedimentos de operação e manutenção básica

Após aplicadas as técnicas de redução dos riscos, deve refazer-se uma nova avaliação dos mesmos, de modo a perceber se os perigos foram ou não reduzidos de forma suficiente ou se foram introduzidos novos riscos. Tal como ilustra a Figura 2, este processo deve ser iterativo até se ter a certeza que todos os perigos de todas as possibilidades foram contabilizados e tratados em conformidade com todas as regras de segurança [6] [12].

2.1.3 Distâncias de segurança

Neste subcapítulo é descrito o processo de posicionamento de equipamentos de proteção eletrosensitivos (por exemplo, barreiras luminosas e *scanners* de área) e de estruturas de proteção (por exemplo, estruturas de proteção e gradeamento) de acordo com as normas ISO 13855 [13] e EN ISO 13857 [14].

Distância mínima de segurança

A distância mínima de segurança deve ser calculada através da equação geral seguidamente apresentada:

$$S = (K \times T) + C \quad (2)$$

Onde,

- S é a distância mínima, em mm.
- K representa a velocidade de aproximação do corpo ou partes do corpo e vem, em mm/s.
- T é o tempo total de paragem.
- C é a distância de penetração, em mm.

A distância de penetração, C , é a distância máxima que se consegue alcançar até à deteção por parte do equipamento de segurança.

O tempo total de paragem, T , pode dividir-se em duas parcelas:

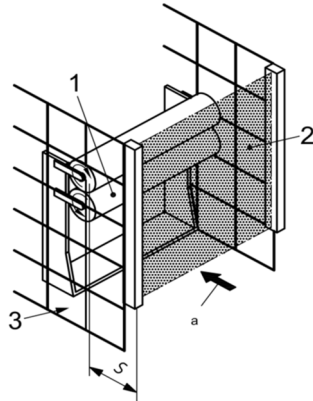
$$T = T_1 + T_2 \quad (3)$$

Onde T_1 corresponde ao tempo máximo entre a atuação do elemento de segurança e a comutação do sinal do *output* e T_2 é o tempo máximo que a máquina ou equipamento perigoso demora a atingir um estado seguro, a partir do momento em que o *output* do elemento de segurança foi comutado.

De seguida, são apresentados os diferentes métodos para o cálculo da distância mínima de segurança para diferentes condições, considerando:

Zona de deteção ortogonal à direção de aproximação

Nesta situação, a altura do feixe mais baixo deve ser no máximo 300mm e o feixe mais alto deve estar a uma altura de, no mínimo, 900mm.



- 1 – Zona perigosa
- 2 – Zona de detecção
- 3 – Barreira fixa
- a – Direção de aproximação
- S – Distância mínima de segurança

Figura 7 - Distância de segurança: zona de detecção ortogonal à direção de aproximação [13].

Para esta situação são considerados 4 casos tipo:

➤ **Caso 1 - $d \leq 40mm$**

Caso em que o sensor do equipamento de proteção tem uma capacidade de detecção $d \leq 40mm$ de diâmetro.

Assim, usa-se a equação geral, Equação 2, sendo os valores dos parâmetros:

$$\begin{aligned} k &= 2000mm/s \\ C &= 8(d - 14), C > 0 \end{aligned} \quad (4)$$

Ou seja,

$$S = (2000 \times T) + 8(d - 14) \quad (5)$$

O valor mínimo não deve ser menor que 100mm.

Contudo, se com os valores acima indicados se obtiver uma distância mínima de segurança for superior a 500mm, o valor de K deverá ser igual a 1600mm/s, ficando:

$$S = (1600 \times T) + 8(d - 14) \quad (6)$$

➤ **Caso 2 - $40mm < d \leq 70mm$**

Caso em que o sensor do equipamento de proteção tem uma capacidade de detecção $d > 40mm$ e $d \leq 70mm$ de diâmetro.

Deve utilizar-se a equação geral com os seguintes valores:

$$\begin{aligned} k &= 1600 mm/s \\ C &= 850 mm \end{aligned}$$

Obtendo-se:

$$S = (1600 \times T) + 850 \quad (7)$$

O valor de 850mm é considerado o valor *standard* do alcance de um braço humano.

➤ **Caso 3 - 2, 3 ou 4 feixes**

Caso em que o sensor do equipamento de proteção possui apenas 2, 3 ou 4 feixes em que apenas garante a detecção do corpo inteiro. A detecção de mãos ou outras partes do corpo não é garantida.

A distância mínima de segurança é calculada recorrendo à mesma equação do caso anterior, equação 7.

➤ **Caso 4 - feixe único**

Caso em que o sensor do equipamento de proteção possui apenas um único feixe.

Deve utilizar-se a equação geral com os seguintes valores:

$$k = 1600mm/s$$

$$C = 1200mm$$

Obtendo-se:

$$S = (1600 \times T) + 1200 \quad (8)$$

É de uso recorrente na indústria o posicionamento do feixe a uma altura de 750mm do solo.

Zona de detecção paralela à direção de aproximação

A altura H é a altura de detecção relativamente ao plano de referência, por exemplo, o solo. Esta deve ser inferior a 1000mm.

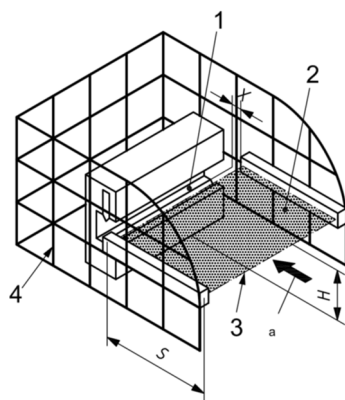
A altura mínima permitida para a zona de detecção H deve ser calculada a partir da seguinte equação:

$$H = 15 (d - 50) , H > 0 \quad (9)$$

Deve utilizar-se a equação geral com os seguintes valores:

$$k = 1600mm/s$$

$$C = (1200 - 0,4 \times H)mm , C > 850mm \quad (10)$$



- 1 – Zona perigosa
- 2 – Zona de detecção
- 3 – Aresta da zona de detecção
- 4 – Barreira fixa
- a – Direção de aproximação
- S – Distância mínima de segurança
- H – Altura de detecção

Figura 8 - Distância de segurança: zona de detecção paralela à direção de aproximação [13].

Zona de detecção inclinada relativamente à direção de aproximação

Nesta situação, se o ângulo entre a zona de detecção e a direção de aproximação for superior a $\pm 30^\circ$ considera-se como perpendicular à direção de aproximação. Caso contrário, considera-se paralela, neste caso, os parâmetros d e H são referentes à aresta de detecção mais afastada da zona perigosa.

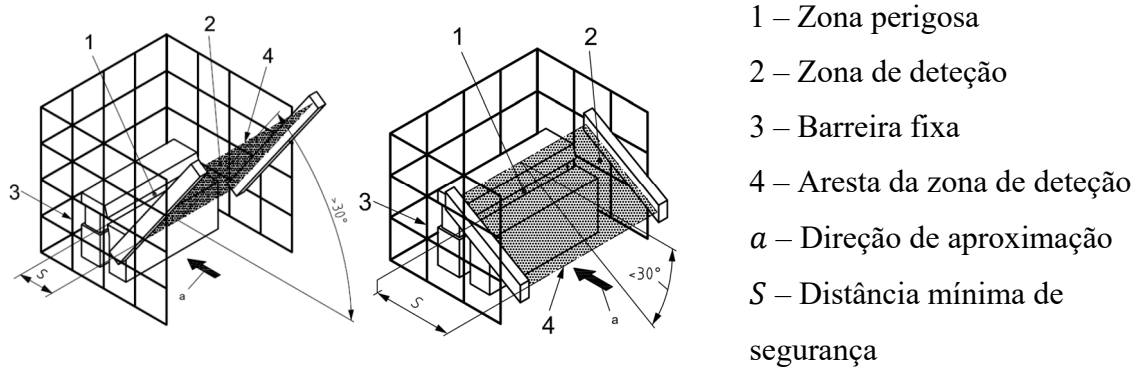


Figura 9 - Distância de segurança: zona de detecção inclinada relativamente à direção de aproximação [13].

Alcance através do contorno do equipamento eletrosensitivo

Nesta situação deve obter-se primeiro o valor mínimo para a distância de segurança considerando o atravessamento na direção de aproximação vertical da zona de detecção que deverá, posteriormente, ser comparado com o valor que se obtiver através abordagem que será apresentada neste subcapítulo.

A distância mínima de segurança é calculada utilizando a seguinte equação:

$$S = (K \times T) + C_{RO} \quad (11)$$

Sendo,

$$K = 2000 \text{ mm/s}$$

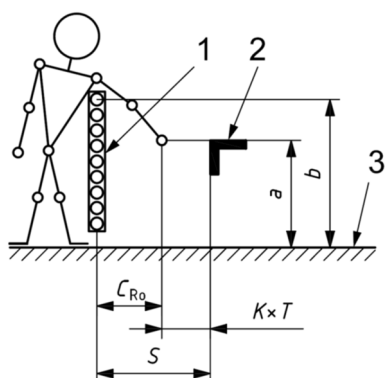
Obtendo-se:

$$S = (2000 \times T) + C_{RO} \quad (12)$$

C_{RO} corresponde à distância adicional dada pela Figura 10 e representa a distância que uma parte do corpo consegue percorrer na direção da zona perigosa até à detecção, por parte do dispositivo de proteção eletrosensitivo.

Se a distância S obtida for superior a 500 mm , o valor de K deverá ser igual a 1600 mm/s , ficando:

$$S = (1600 \times T) + C_{RO} \quad (13)$$



- 1 – Equipamento eletrosensitivo
 2 – Zona perigosa
 3 – Plano de referência
a – Altura da zona perigosa
b – Altura da aresta superior do equipamento eletrosensitivo
S – Distância mínima de segurança
 C_{RO} – Distância adicional

Figura 10 - Distância de segurança: alcance através do contorno do equipamento eletrosensitivo [10].

A Tabela 9 é uma tabela de dupla entrada que relaciona a altura do ponto perigoso com a altura da aresta de detecção mais alta do dispositivo de proteção eletrosensitivo, de modo a obter a distância adicional a considerar [13].

Tabela 9 - Distância de segurança: distância adicional para a situação do alcance através do contorno do equipamento eletrosensitivo [13]

Dimensions in millimetres

Height of hazard zone <i>a</i>	Height of upper edge of the detection zone of the electro-sensitive protective equipment <i>b</i>											
	900	1 000	1 100	1 200	1 300	1 400	1 600	1 800	2 000	2 200	2 400	2 600
	Additional distance to hazard zone C_{RO}											
2 600 ^a	0	0	0	0	0	0	0	0	0	0	0	0
2 500	400	400	350	300	300	300	300	300	250	150	100	0
2 400	550	550	550	500	450	450	400	400	300	250	100	0
2 200	800	750	750	700	650	650	600	550	400	250	0	0
2 000	950	950	850	850	800	750	700	550	400	0	0	0
1 800	1 100	1 100	950	950	850	800	750	550	0	0	0	0
1 600	1 150	1 150	1 100	1 000	900	850	750	450	0	0	0	0
1 400	1 200	1 200	1 100	1 000	900	850	650	0	0	0	0	0
1 200	1 200	1 200	1 100	1 000	850	800	0	0	0	0	0	0
1 000	1 200	1 150	1 050	950	750	700	0	0	0	0	0	0
800	1 150	1 050	950	800	500	450	0	0	0	0	0	0
600	1 050	950	750	550	0	0	0	0	0	0	0	0
400	900	700	0	0	0	0	0	0	0	0	0	0
200	600	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0

Alcance através do contorno de estruturas de proteção

Nas situações onde a separação entre o perigo e o Homem é feita por uma barreira física, esta deve ser colocada a uma distância de segurança de acordo com a norma EN ISO 13857.

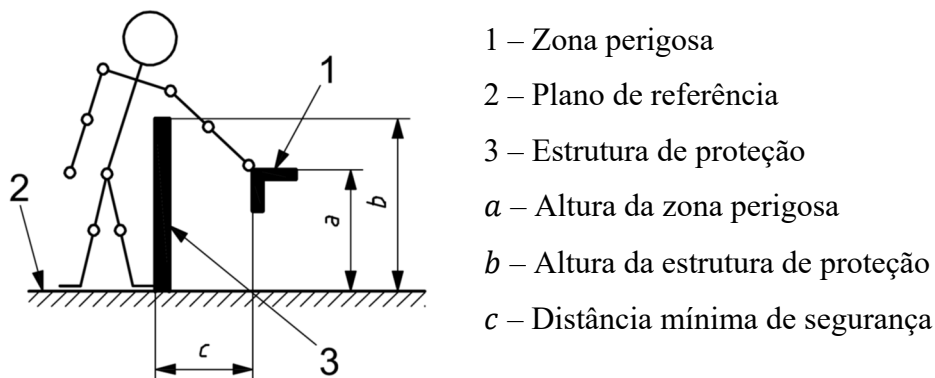


Figura 11 - Distância de segurança: alcance através do contorno de estruturas de proteção [14].

Esta norma dita que para prevenir o alcance do perigo através do contorno de uma estrutura de proteção, esta deve estar a uma distância do perigo, c , dada pela Tabela 10 ou pela Tabela 11.

A Tabela 10 diz respeito a situações onde o risco inerente é baixo e a Tabela 11 a situações de risco mais elevado. À semelhança do subcapítulo anterior, estas tabelas são de dupla entrada, em que a distância mínima de segurança, c , pode ser obtida em função da altura da estrutura de proteção, b , e da altura a que o elemento perigoso se encontra do plano de referência, a .

Tabela 10 - Distância de segurança: distância para a situação do alcance através do contorno de uma estrutura de proteção para riscos reduzidos [14]

Dimensions in millimetres

Height of hazard zone <i>a</i>	Height of protective structure <i>b</i>								
	1 000	1 200	1 400	1 600	1 800	2 000	2 200	2 400	2 500
	Horizontal safety distance to hazard zone, <i>c</i>								
2 500	0	0	0	0	0	0	0	0	0
2 400	100	100	100	100	100	100	100	100	0
2 200	600	600	500	500	400	350	250	0	0
2 000	1 100	900	700	600	500	350	0	0	0
1 800	1 100	1 000	900	900	600	0	0	0	0
1 600	1 300	1 000	900	900	500	0	0	0	0
1 400	1 300	1 000	900	800	100	0	0	0	0
1 200	1 400	1 000	900	500	0	0	0	0	0
1 000	1 400	1 000	900	300	0	0	0	0	0
800	1 300	900	600	0	0	0	0	0	0
600	1 200	500	0	0	0	0	0	0	0
400	1 200	300	0	0	0	0	0	0	0
200	1 100	200	0	0	0	0	0	0	0
0	1 100	200	0	0	0	0	0	0	0

Tabela 11 - Distância de segurança: Distância para a situação do alcance através do contorno de uma estrutura de proteção para riscos elevados [11]

Dimension in millimetres

Height of hazard zone <i>a</i>	Height of protective structure <i>b</i>									
	1 000	1 200	1 400	1 600	1 800	2 000	2 200	2 400	2 500	2 700
	Horizontal safety distance to hazard zone, <i>c</i>									
2 700	0	0	0	0	0	0	0	0	0	0
2 600	900	800	700	600	600	500	400	300	100	0
2 400	1 100	1 000	900	800	700	600	400	300	100	0
2 200	1 300	1 200	1 000	900	800	600	400	300	0	0
2 000	1 400	1 300	1 100	900	800	600	400	0	0	0
1 800	1 500	1 400	1 100	900	800	600	0	0	0	0
1 600	1 500	1 400	1 100	900	800	500	0	0	0	0
1 400	1 500	1 400	1 100	900	800	0	0	0	0	0
1 200	1 500	1 400	1 100	900	700	0	0	0	0	0
1 000	1 500	1 400	1 000	800	0	0	0	0	0	0
800	1 500	1 300	900	600	0	0	0	0	0	0
600	1 400	1 300	800	0	0	0	0	0	0	0
400	1 400	1 200	400	0	0	0	0	0	0	0
200	1 200	900	0	0	0	0	0	0	0	0
0	1 100	500	0	0	0	0	0	0	0	0

2.2 Sistemas HRC

Em sistemas HRC, *Human Robot Collaboration* ou Colaboração Homem Robô, o robô assiste o operador nas suas tarefas, não o substituindo, mas complementando as suas capacidades. Estes sistemas são característicos das Fábricas do Futuro, as Factory 4.0 [15]

Existem diferentes formas de automação dependendo da forma como o Homem e a máquina interagem. Sendo o objetivo a completa colaboração, Factory 4.0, ou seja, Homem e Robô a partilhar o mesmo espaço e a realizar tarefas em conjunto, os primeiros passos para atingir este objetivo podem passar pelo desenvolvimento e separação de diferentes situações: colaboração, cooperação e coexistência [16].

De seguida, são apresentados exemplos práticos de cada uma destas situações[16].

Colaboração

Nesta situação o robô agarra numa peça do tapete transportador na traseira da célula, transporta-a para junto do operador onde este lhe acrescenta uma parte. Adicionalmente, a bancada onde o robô se encontra é móvel e pode ser deslocada para outra zona da fábrica. A Figura 12 ilustra esta situação.

Para garantir a segurança do operador a célula dispõe de *scanners* de área a toda a volta do robô, que adapta as características de movimento de robô conforme a proximidade do operador.



Figura 12 - Exemplo de colaboração entre o operador e o robô [16].

Cooperação

A situação descrita pela Figura 13 é uma estação de transferência onde o operador coloca numa bancada módulos pré-montados para uma máquina ferramenta. O robô é responsável pela colocação deste módulo na mesa da máquina ferramenta.

Como equipamentos de segurança têm-se: barreiras físicas nas laterais da célula e um *scanner* de área, que monitoriza o espaço em frente à bancada. Mais uma vez, o robô ajusta as suas características de movimento em função da proximidade do operador.

Este exemplo difere do anterior por não existir contacto direto com o robô



Figura 13 - Exemplo de cooperação entre o operador e o robô [16].

Coexistência

Neste exemplo, o robô é responsável por fazer o *pick and place* de componentes de um tapete transportador para uma mesa rotativa. Do outro lado da mesa é a zona de trabalho do operador que é protegida por duas barreiras luminosas, uma horizontal e outra vertical.

A mesa rotativa é o elemento perigoso, uma vez que existe uma estrutura que separa a área de operação do robô e a área de trabalho do operador.

A barreira luminosa vertical deteta a entrada do operador na célula e provoca a paragem da rotação da mesa. A barreira horizontal faz a verificação da existência do operador ou de algum objeto dentro do espaço protegido.

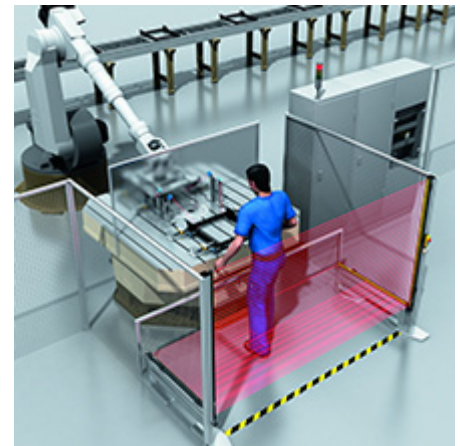


Figura 14 - Exemplo de coexistência entre o operador e o robô [16].

2.2.1 Requerimentos para operações colaborativas

Segundo as normas ISO 10218 [17] e ISO/TR 15066, as operações colaborativa devem ser sinalizadas visualmente e são divididas em 4 categorias [17] [18] [19]:

- *Safety-rated monitored stop*
- *Hand guiding*
- *Speed and separation monitoring*
- *Power and force limiting by inherent design or control*

Safety-rated monitored stop

O robô deve permanecer imóvel se o operador se encontrar no espaço de trabalho colaborativo. Este pode retomar a operação automática assim que o operador se afastar.

Hand guiding

Possibilita que o operador controle diretamente o robô dentro da sua área de trabalho.

A interface de controlo deve estar situada perto do robô (*end-effector*) e estar equipada com os seguintes elementos:

- Botoneira de emergência
- Botão *enabling* de três posições

Speed and separation monitoring

Os parâmetros de movimento do robô são ajustados em função da proximidade do operador. Se a distância mínima for ultrapassada o robô deve ser parado.

Power and force limiting by inherent design or control

Os valores máximos para os parâmetros de movimentação do robô, potência, velocidade, binário e força estática, devem ser definidos pelo *Risk Assessment* e tais não podem ser ultrapassados.

2.3 Sistemas para rotação de vigas

De seguida, são apresentados os equipamentos para rotação de vigas mais usados na indústria atualmente.

Estes equipamentos servem para substituir/diminuir o uso de pontes rolantes para realizar a rotação de vigas de maiores dimensões ou até mesmo de estruturas metálicas.

2.3.1 FastRotator

O FastRotator é um equipamento desenvolvido pela empresa Portuguesa Lomboser SA e concebido, especificamente, para auxiliar o trabalho de rotação e posicionamento na soldadura de estruturas metálicas [20].

As imagens seguintes, Figura 15 e Figura 16, apresentam dois modelos Fastrotator, ambos de elevada capacidade.



Figura 15 - FastRotator FR2012: capacidade 12 ton., rotação 90° em 30s, diâmetro de viragem máximo de 2800mm [20].



Figura 16 - FastRotator FR1508: capacidade 8 ton., rotação 180° em 18s, diâmetro de viragem máximo de 2100mm [20].

Este equipamento é composto por duas estruturas iguais, sendo que em cada uma delas existe uma cinta de carga. No topo de cada estrutura existe uma polia que através da respetiva cinta imprime rotação na viga.

Toda a parte superior da estrutura pode ser elevada hidraulicamente de modo a acomodar vigas ou estruturas de diversas dimensões. Este equipamento é normalmente acompanhado de cavaletes de apoio, que servem para apoiar a viga e fechar o circuito da máquina de soldar.

Seguidamente, são apresentadas algumas vantagens e desvantagens do FastRotator relativamente aos equipamentos de rotação de vigas.

Vantagens

- Flexibilidade, pode acomodar desde simples vigas a estruturas completas
- Elevada capacidade de carga (40 ton. [20])
- Diversidade de modelos – várias cargas
- Roda a viga rapidamente (ordem dos 5s para uma rotação de 90°)
- Concebido especificamente para soldadura

Desvantagens

- Elevada ocupação de espaço, quer em altura quer em área de chão de fábrica
- Carregamento e descarregamento de vigas dificultado pela própria estrutura
- Vocacionado para auxiliar em processo de soldadura manual
- Necessidade de preparar a viga para rotação, colocação de cintas

2.3.2 Viradores de corrente

Estes equipamentos são os mais comuns no mundo fabril e diferenciam-se por suportar a viga através de uma corrente metálica conectada a dois pórticos simétricos.

A figura seguinte, Figura 17, ilustra uma máquina desta categoria.



Figura 17 – Virador de corrente: Stierli-Bieger Rotator 1000, cap. 12 ton., rotação 90° em 15s, diâmetro de viragem máximo de 1000mm.

Na extremidade de cada um dos pórticos encontra-se um braço articulado, que permanece vertical quando não existe nenhuma viga sobre a corrente. Estes braços tomam a posição ilustrada na figura anterior de modo automático assim que se baixa a viga. Ou seja, o próprio peso da viga fecha a estrutura.

Cada extremidade da corrente está, no interior da estrutura, conectada a um sistema mecânico. No caso da figura acima está conectada ao respetivo motor elétrico com redutor.

A rotação da viga é conseguida pelo acionando dos motores elétricos que enrolam ou desenrolam a corrente.

Relativamente aos outros equipamentos de rotação de vigas apresentados nesta dissertação, os viradores de corrente têm as seguintes vantagens e desvantagens [21].

Vantagens

- A própria corrente fecha o circuito da máquina de soldar
- Flexibilidade e ergonomia no posicionamento da viga
- Fácil carregamento e descarregamento de vigas
- Não existe a necessidade de passar cintas ou correntes adicionais
- Robusto e durável

Desvantagens

- Secção da viga limitada ao tamanho do equipamento
- Acessibilidade reduzida na zona dos pórticos
- Vocacionado para auxiliar em processo de soldadura manual

2.3.3 Ocean Flipper

O Ocean Flipper é um produto da Ocean Machinery que foi inicialmente desenvolvido para auxiliar a rotação e o posicionamento de vigas em máquinas de corte e furação. Contudo, pode ser utilizado noutros processos, tais como soldadura, pintura ou *transfer* de componentes [22].

A integração desta solução requer pelo menos um par de Ocean Flippers. A figura seguinte, Figura 18, ilustra apenas um Ocean Flipper.



Figura 18 – Ocean Flipper: capacidade 12 ton., rotação 90° em 15s, dimensão máxima da viga de 1000mm [22].

Cada Ocean flipper é composto por duas pás, cada uma controlada por um cilindro hidráulico. O conjunto das duas pás pode ainda ser elevado (verticalmente) e transladado (horizontalmente), sendo controlados, ambos os movimentos, por sistemas hidráulicos. O movimento horizontal do segundo Ocean Flipper (par correspondente) é realizado através de um veio de transmissão que garante o sincronismo do movimento.

O Ocean Flipper, relativamente aos outros equipamentos apresentados nesta dissertação, apresenta as seguintes vantagens e desvantagens [21].

Vantagens

- Compacidade do sistema, baixa ocupação de espaço
- Precisão de posicionamento
- Rapidez nos movimentos
- Fácil carregamento e descarregamento de vigas
- Não existe a necessidade de passar cintas ou correntes adicionais

Desvantagens

- Secção da viga limitada ao tamanho do equipamento
- Menor capacidade de carga

2.4 SoftPLC

Nas primeiras máquinas com controlo automático, a automação era conseguida através da combinação de simples dispositivos, nomeadamente botões, relés, temporizadores e contadores. Por outras palavras, a automação era baseada em sistemas de lógica *hard-wired*.

Como é possível imaginar, o aumento da complexidade destes sistemas levava a um aumento considerável da complexidade do sistema elétrico e, uma vez que não era possível testar antecipadamente os programas, as tarefas de deteção de erros ou mesmo de manutenção do equipamento tornavam-se cada vez mais difíceis [23].

Os PLC's ou *Programmable Logic Controlers* são dispositivos que permitem, realizar o “*software*” de todos os dispositivos de forma flexível e controlada. Cada PLC é composto por, pelos menos, um CPU responsável pelo varrimento de um programa, monitorizando, assim, uma série de *inputs* e manipulando, através de lógica programável, o valor dos seus *outputs* [24]. Estes equipamentos substituíram, assim, a lógica *hard-wired* até aqui utilizada.

Com o contínuo desenvolvimento das tecnologias dos semicondutores, dos circuitos integrados e dos microprocessadores, a utilização do PLC espalhou-se por toda a indústria. Atualmente, os PLC permitem todo o tipo de controlos, desde simples operações lógicas a operações mais complexas como a implementação de controladores PID [23].

SoftPLC é um *software* que visa simular o comportamento de um PLC convencional através de um computador. Este tipo *software* é cada vez mais usado na indústria de forma a combater as atuais desvantagens dos PLC tradicionais [25].

2.4.1 SoftPLC vs PLC convencional

Neste subcapítulo é feita uma análise comparativa das vantagens dos *soft*PLC e dos PLC tradicionais [25] [26] [27] [28].

Características operacionais

Real-Time Operating System (RTOS) é um conceito de sistema operativo (*Operating System* – OS) que visa garantir que cada tarefa é executada num certo intervalo de tempo, garantindo sincronismo e características determinísticas na execução em tempo real de um programa e repetibilidade de comportamento.

Os PLC tradicionais utilizam sistemas operativos deste tipo, sendo, portanto, ótimos para o controlo de processos mais críticos. Têm, também, uma unidade de processamento dedicada à execução do programa de controlo, enquanto que num *soft*PLC muitas vezes o poder de processamento é partilhado com outras tarefas do OS do computador.

Assim, os PLC tradicionais conseguem ter tempos de resposta mais curtos para programas mais simples. Já os *soft*PLC têm normalmente ao seu dispor maior poder de processamento e maior memória, adequado a programas maiores e mais complexos.

Robustez

Neste campo o PLC convencional tende a ser superior à maior parte dos *soft*PLC.

O PLC convencional não tem partes móveis e é concebido exatamente para ser durável e resistir a todas adversidades de um ambiente fabril hostil. No entanto, já existem computadores habilitados para resistir a ambientes industriais.

Hardware

Os PLC convencionais já incluem, normalmente, portas IOs e capacidades de comunicação compatíveis com as redes de campo mais utilizadas. É uma solução que não necessita de praticamente de nenhum equipamento adicional para o seu funcionamento. Para além disso, são normalmente modulares, conferindo ao utilizador alguma flexibilidade na escolha do equipamento.

Relativamente aos *soft*PLC, estes requerem, no mínimo, os seguintes componentes de *hardware*:

- Computador
- Módulos de IO remotos

A necessidade de equipamento e investimento adicional é uma desvantagem. Contudo, os *soft*PLC beneficiam das características do computador que os suporta, quer na capacidade de comunicação alargada quer na capacidade gráfica do mesmo.

Assim, apesar dos PLC convencionais saciarem as principais necessidades industriais em termos de comunicação, os *soft*PLC tendem a suportar uma maior variedade de protocolos e redes de comunicação. Os *soft*PLC possibilitam ainda comunicar em simultâneo, usando diferentes protocolos e vias de comunicação, abrindo o leque de opções do *hardware* que se pode integrar.

Segurança - Security

Naturalmente, o PLC convencional está menos exposto a vírus, *malwares* e ao acesso por elementos não autorizados, principalmente pela utilização de um OS dedicado.

Já os *soft*PLC, por serem integrados num computador, são mais suscetíveis a falhas de segurança. Neste aspeto, o elevado poder de comunicação constitui uma desvantagem. O recurso a software antivírus ou *antimalware* é um exemplo de aplicações que com as quais o *soft*PLC vai ter que partilhar os recursos do computador.

Ambiente e características de programação

Na sua generalidade, as linguagens de programação utilizadas pelos fabricantes de PLC são idênticas, porém, existem, muitas vezes, incompatibilidades quer ao nível da configuração, dos métodos de mapeamento utilizados ou mesmo ao nível de estrutura gramatical.

No entanto, com o aparecimento da norma IEC61131-3, os fabricantes começam a adotar arquiteturas standardizadas. O conteúdo desta norma será abordado no seguinte subcapítulo.

Os *soft*PLC, pela sua proeminência nos dias de hoje, seguem estes standards, usufruindo de todas suas vantagens na totalidade, destacando-se:

- Portabilidade e compatibilidade de programas entre sistemas de marcas diferentes.
- Permite uma melhor organização do código, como por exemplo, encapsulamento de funções ou porções de código.
- Variedade de linguagens standardizadas compatíveis entre si.
- Facilidade na monitorização, na simulação e no *debug* dos programas.

2.4.2 Norma IEC 61131-3

Antes do aparecimento desta norma, cada fabricante tinha a sua própria linguagem e os seus próprios métodos de programação. A linguagem *ladder* era uma das mais populares pela sua simplicidade e facilidade de compreensão, porém, cada fabricante usava o seu próprio “dialecto”, resultando, mais uma vez, na não transversalidade da mesma.

A norma IEC 61131 (denominada IEC 1131 até 1998) aparece em 1993 e está dividida em várias partes, visando padronizar o sistema de programação e o *hardware* dos PLC.

IEC 61131-3 é a parte relativa aos métodos de programações dos PLC e apresenta, de forma estruturada, as regras e os elementos de programação para algumas linguagens [23] [29].

De seguida, são apresentadas as principais características desta norma (IEC 61131-3) [29], alguns conceitos importantes para a compreensão do sistema de programação [30] e uma breve apresentação das linguagens padronizadas [30][31].

Principais características

- Facilidade de organização e estruturação dos programas. Permite que o projeto de PLC, programa do PLC, seja dividido em elementos funcionais designados *Program Organisation Units* (POUs) e organizados segundo uma hierarquia definida.
- O ambiente de edição deve ser capaz de corrigir ou detetar erros de sintaxe à medida que o programa é escrito, evitando, assim, uma das principais fontes de erro.
- Ferramentas que permitem a gestão de diferentes programas, ou tarefas, para que cada programa possa ser executado independentemente dos outros, com periodicidades diversas, tempos de ciclo limitados e até mesmo em paralelo uns com os outros. Isto é conseguido atribuindo as diferentes partes do programa a diferentes tarefas.
- Suporte para uma linguagem sequencial, *Sequential Function Chart* (SFC) que, graficamente, descreva um ou mais processos, independentes ou não, através de etapas, ações e transições entre etapas.
- Possibilidade de criar *data structures*, dotando assim as linguagens de PLC com funcionalidades típicas das linguagens de alto nível mais comuns (C, Java, Pascal, PHP, Visual Basic).
- Descreve 5 linguagens de programação, 3 gráficas e 2 escritas. O utilizador tem a flexibilidade de poder escolher e utilizar qualquer uma destas para cada parte do programa.
- As linguagens e métodos de programação, que por esta norma são padronizados, pretendem fornecer todas as ferramentas para que os PLC alcancem o estatuto de sistemas abertos, acabando assim com linguagens e métodos de programação proprietários e, consequentemente, visando a total compatibilidade entre produtos de marcas diferentes. Um produto dito de sistema aberto é aquele que adota padrões e técnicas, de modo que a sua integração com outros sistemas abertos seja facilitada e não apresente incompatibilidades.

Conceitos importantes

➤ Program Organization Unit - POU

Correspondem a pequenas unidades, ou blocos independentes, resultantes do encapsulamento de *software*. Esta independência facilita a modularização dos projetos e promove a reutilização de *software*. Um projeto de PLC é constituído por diversas POUs.

Podem ser de três tipos: *Function* (FUN), *Function Block* (FB) ou *Program* (PROG).

Function ou Função é um elemento sem memória. Assim, quando é chamada produz sempre os mesmos resultados para os mesmos *inputs*. Podem ter diversos *inputs* e *outputs*, mas apenas um *function value*. *Function value* corresponde ao valor que é retornado como resultado da função

Function Block ou bloco de funções difere da função por possuir memória, ou seja, os seus *outputs* dependem, não só, dos parâmetros de entrada, mas também do seu estado interno.

Program ou Programa, para além de ter as mesmas características que o bloco de funções, permite a declaração e o acesso às variáveis físicas correspondentes às portas IO do PLC

A figura seguinte, Figura 19, ilustra as hierarquias de chamada dos vários tipos de POU's.

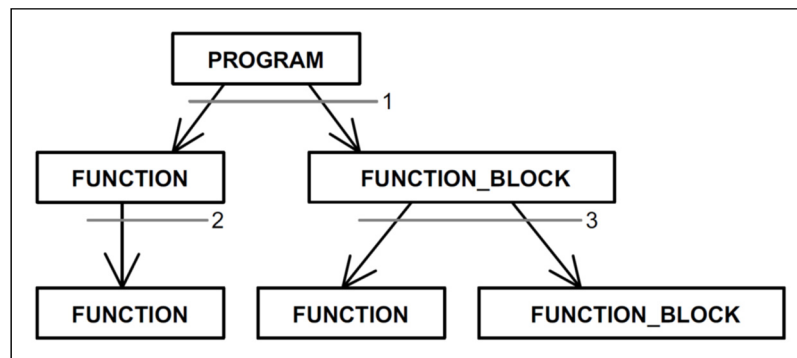


Figura 19 - Hierarquias de chamada das POU [30].

- Programas podem chamar blocos de funções e/ou funções.
- Blocos de funções podem chamar funções e/ou outros blocos de funções.
- Funções apenas podem chamar outras funções.

É de notar, também, que uma POU não se pode chamar a ela própria.

➤ Estrutura de programação de PLCs

A uniformidade da arquitetura de *software* dos PLC é crucial para a portabilidade das aplicações entre PLC de diferentes marcas.

Os elementos da configuração descrevem as propriedades do *run-time program*, os canais de comunicação e de integração do *hardware* do PLC. A interligação dos elementos de configuração com os elementos do PLC dependem da arquitetura de *hardware* do PLC.

A figura seguinte, Figura 20, ilustra a arquitetura de *software* do PLC segundo a norma IEC 61131.

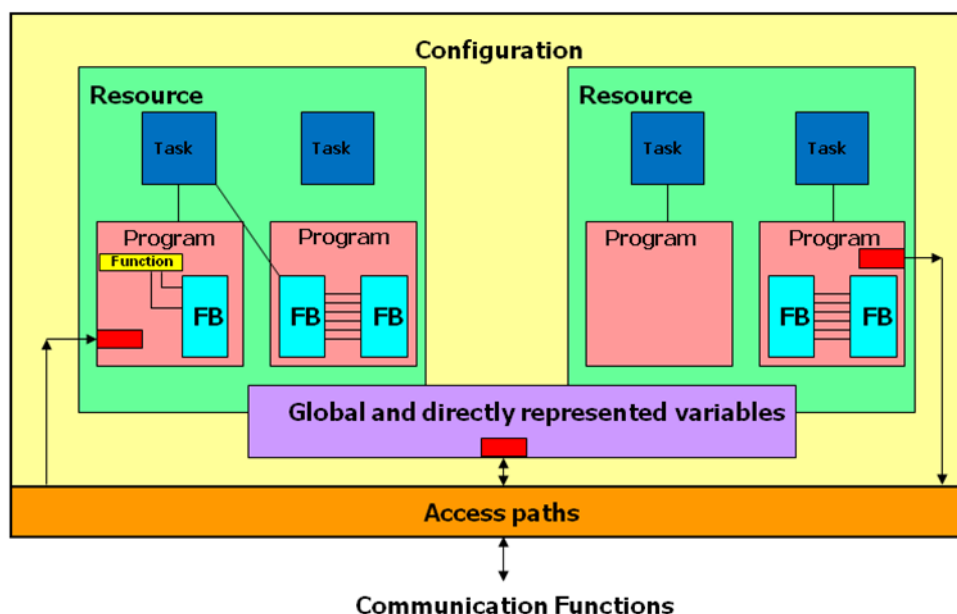


Figura 20 - Modelo de *software* do PLC de acordo com a norma IEC 61131 [32].

Configuration ou Configuração: agrupa todos os recursos de um PLC e providencia-os com meios para troca de dados. Por exemplo, define variáveis globais e vias de comunicação entre as várias configurações.

Resources ou Recursos: é representante de uma ou mais unidades de processamento (*Central Processing Unit* – CPU) do PLC, às quais estão associadas tarefas. Define variáveis globais e cede *inputs* e *outputs* ao *run-time program*.

Tasks ou Tarefas: atribui propriedades de *run-time* às POU's, ou seja, faz uma gestão das POU's que lhe estão associadas, invocando-as com tempos de ciclo, periodicidade e prioridade bem definidos. A associação das tarefas com as POU's define as propriedades do *run-time program*. *Run-time program* é um programa capaz de correr de forma independente num CPU e guardar todas as propriedades de *run-time* de cada conjunto tarefa-POU.

A figura seguinte, Figura 21, é uma representação gráfica da declaração de tarefas.

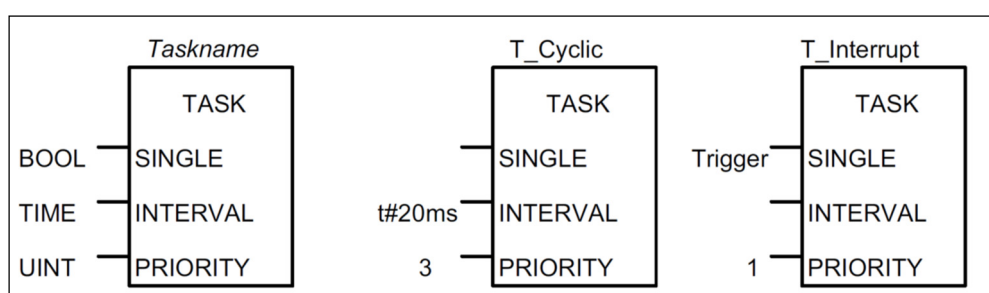


Figura 21 - Exemplo de representações gráficas de declarações de tarefas. À esquerda é representado um bloco genérico, ao centro um bloco cíclico e à direita um bloco não cíclico. Esta representação gráfica não é contemplada pela norma IEC 61131 [30].

O bloco da esquerda é genérico e mostra todas as propriedades de uma tarefa. Existem 3 *inputs*:

- *Single* – aceita uma variável booleana. Aquando da *rising edge* desta variável a POU associada a esta tarefa é executada uma vez.
- *Interval* – aceita uma variável de tempo. Se este *input* for diferente de zero a tarefa será executada periodicamente com período igual ao valor do *input*.
- *Priority* – aceita um inteiro. O valor deste *input* define a prioridade desta tarefa relativamente às outras tarefas.

O bloco central e o bloco da direita são exemplos de possíveis tarefas. O bloco central declara uma tarefa cíclica, de período igual 20ms e com prioridade 3. O bloco mais à direita declara uma tarefa que será executada aquando da *rising edge* da variável booleana “trigger” com prioridade igual 1.

Linguagens padronizadas

➤ Lista de Instruções

Instruction List ou Lista de Instruções (IL) é uma linguagem semelhante à linguagem *assembly*. Cada comando do PLC é descrito por apenas uma linha.

A figura seguinte, Figura 22, apresenta alguns exemplos de instruções em IL.

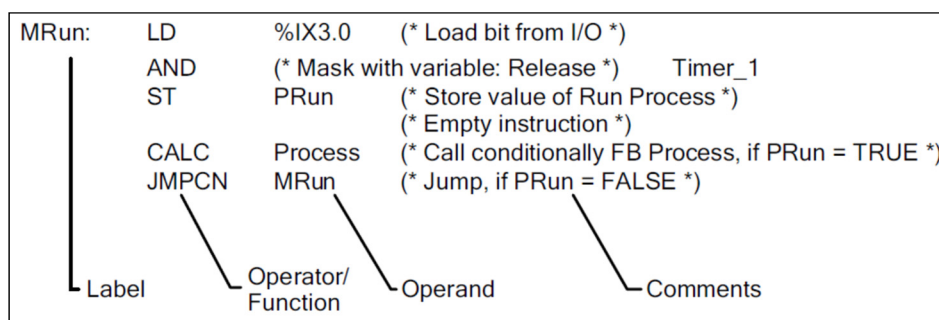


Figura 22 - Exemplo de instruções em IL [30].

➤ Texto Estruturado

Structured Text ou Texto Estruturado (ST) é uma linguagem dita de alto nível comparável com linguagens como PASCAL e C.

É composta por séries de instruções capazes de descrever funcionalidades complexas de maneira compacta. Contém todos os elementos essenciais das linguagens modernas, como por exemplo *If-Else-Then*, *Case of*, *For*, *While*, *Repeat*, entre outros.

Uma das principais utilizações é na definição de blocos lógicos, ou seja, o código complexo é descrito em texto estruturado que depois é encapsulado dentro de um bloco de funções. Posteriormente, este bloco pode ser usado por qualquer uma das linguagens por esta norma definidas.

A figura seguinte, Figura 23, apresente alguns exemplos de instruções em ST.

```

I:=25;
WHILE J<5 DO
    Z:= F(I+J);
END_WHILE

IF B_1 THEN
    %QW100:= INT_TO_BCD(Display)
ENDIF

CASE TW OF
    1,5: TEMP := TEMP_1;
    2:   TEMP := 40;
    4:   TEMP := FTMP(TEMP_2);
ELSE
    TEMP := 0;
    B_ERROR :=1;
END_CASE
    
```

Figura 23 - Exemplo de instruções em ST [12].

➤ Diagrama de blocos

Function Block Diagram ou Diagrama de Blocos (FBD) é, também, uma linguagem popular entre os programadores de PLCs.

Permite interligar graficamente funções, blocos de funções e programas com os blocos lógicos convencionais, como operadores lógicos, memórias, entre outros. O comportamento do programa é conseguido através do condicionamento de sinal entre todos os blocos de processamento.

A figura seguinte, Figura 24, apresenta alguns exemplos de elementos de um FBD.

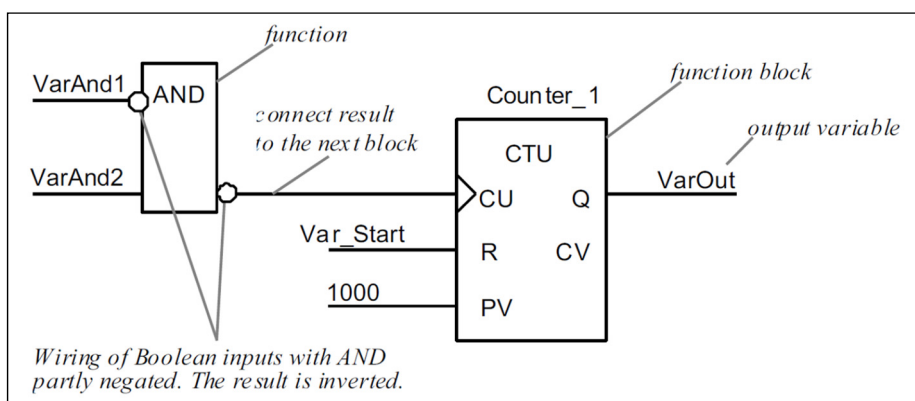


Figura 24 - Exemplo de elementos de um FBD [12].

➤ Diagrama de contactos

Ladder Diagram ou Diagrama de Contactos (LD) é, como já referido no início deste subcapítulo, a linguagem mais popular de programação de PLCs.

É uma linguagem muito simples, trabalha maioritariamente com valores booleanos e tem um funcionamento semelhante aos esquemas elétricos comuns.

A figura seguinte, Figura 25, apresenta alguns exemplos de elementos de um LD.

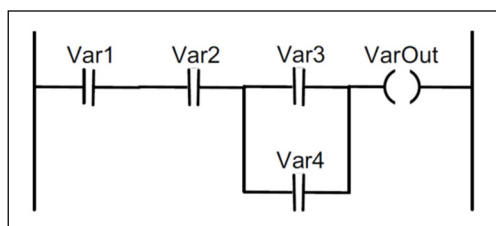


Figura 25 - Exemplo de elementos de um LD [12].

➤ Diagrama de estados

Sequential Function Chart ou Diagrama de Estados (SFC) é uma linguagem que pode ser descrita graficamente e textualmente e que deriva das metodologias Petri Nets e Grafcet.

Esta linguagem decompõe um processo numa série de passos mais pequenos e controláveis, isto é, consegue descrever programas complexos através uma sequencia lógica de etapas. Assim, torna-se fácil a descrição de processos sequenciais e/ou paralelos.

Cada etapa representa o estado do sistema naquele momento e tem, normalmente, ações a ela associadas. As transições têm associada uma condição que quando verificada desativa a etapa anterior e ativa a etapa seguinte, obtendo-se, assim, um comportamento sequencial e organizado.

O diagrama de estados é uma linguagem particularmente interessante para a representação global de projetos complexos e de maiores dimensões pois permite a visão da globalidade do projeto de maneira simples e organizada.

A figura seguinte, Figura 26, apresenta um simples exemplo de um programa em SFC.

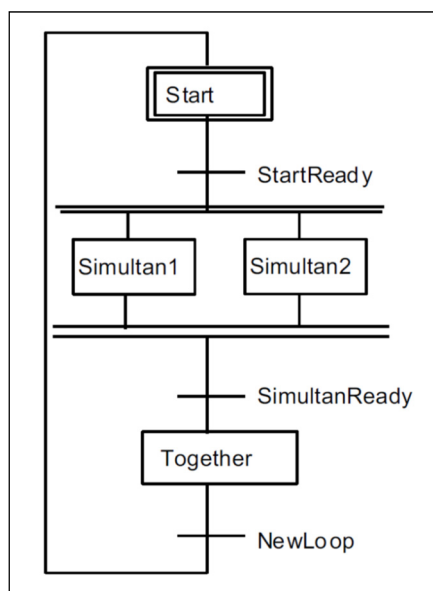


Figura 26 - Exemplo de um programa em SFC [12].

2.5 Interface gráfica - WPF

O WPF, *Windows Presentation Foundation*, é um componente do Microsoft.NET Framework 3.5, que é utilizado para a criação de UIs, *User Interfaces*, permitindo a criação de aplicações de elevado grau de personalização.

O WPF separa a aparência da interface do utilizador do comportamento. A aparência no WPF é criada, geralmente, no XAML (*Extensible Application Markup Language*), o desempenho/comportamento é implementado numa linguagem de programação, como por exemplo, o C#. Ambas as partes estão ligadas entre si através dos eventos de ligação a dados e comandos.

A separação da aparência do comportamento apresenta as seguintes vantagens: personalização dos modelos e das aplicações e permite que as ferramentas de *design* trabalhem em documentos XML “simples”, em vez analisar código. Assim, graças a esta separação, a aparência no WPF é extremamente personalizável, ou seja, é extremamente simples realizar a alteração da aparência de um botão, Figura 27.



Figura 27 - WPF: Alteração da aparência dos botões [33].

Todas as medidas em WPF são unidades lógicas, ou seja, não são pixels. Uma unidade lógica corresponde a 1/96 de uma polegada. Assim, se a resolução do ecrã aumentar, a interface do utilizador permanece inalterável, apenas fica mais nítida. Por outras palavras, com o WPF a interface acompanha o aumento da resolução, sendo independente desta.

A Figura 28 dá o exemplo do que não acontece com o WPF, onde a mesma CheckBox, a duas resoluções diferentes, apresenta tamanhos diferentes.



Figura 28 - WPF: Exemplo onde a aparência é dependente da resolução [33].

Tudo o que é construído no XAML também pode ser feito no código. XAML é, portanto, uma outra maneira de criar e inicializar objetos.NET com relações hierárquicas. Pode utilizar-se WPF sem recorrer ao XAML [33].

As principais vantagens do XAML são [33]:

- Código curto e de fácil leitura
- Separação do código de *design* da lógica da interface do utilizador

A grande diferença entre WinForms e WPF é o facto de que, enquanto o WinForms é simplesmente uma camada em cima dos controlos padrão do Windows, como por exemplo a TextBox, o WPF é construído a partir do zero e não depende dos controlos padrão do Windows em quase todas as situações.

A flexibilidade do WPF, por um lado, facilita a criação de conteúdos inovadores. No entanto, por outro lado, pode dificultar a criação de elementos mais convencionais, como por exemplo: a criação de uma ListView contendo uma imagem e algum texto alinhados. Neste

caso, embora seja relativamente fácil de fazer com o WPF, utilizando o WinForm com o ListViewItem, é feito numa linha de código [34].

A lista que se segue apresenta as principais vantagens para WPF e WinForms [34].

WPF

- Criação de interfaces com aspeto mais moderno e fora do convencional
- Flexibilidade
- O XAML facilita a criação e edição quando o trabalho é dividido entre um designer (XAML) e um programador (C#, VB.NET)
- *Data binding* – separação mais limpa de dados e *layout*
- Usa aceleração de *hardware* para efeitos gráficos – melhor desempenho
- Permite construir interfaces para Windows e aplicações Web

WinForms

- Mais antigo – testado e aprovado
- Maior diversidade de funcionalidades disponíveis no mercado
- O Visual Studio permite criar WinForms de forma intuitiva e rápida

3 Célula robótica - Arquitetura

Neste capítulo encontra-se toda a informação acerca da célula para a sua compreensão e necessária para o início desta dissertação. Assim, primeiramente, são descritos todos os componentes principais e é apresentada uma versão básica do seu *layout*.

Por fim, são descritos todos os fluxos de informação e arquitetura de todo o sistema.

3.1 Funcionamento previsto

Esta célula robótica tem como finalidade a soldadura de vigas metálicas utilizando um robô em colaboração com o operador.

O robô está suspenso num semipórtico que, por sua vez, está montado num eixo linear capaz de o movimentar com um curso de 13m. Para além do robô, também está a bordo do pórtico uma máquina de soldar, adaptada ao robô, e um projetor laser.

Inicialmente, o projetor laser projeta na viga a posição dos componentes a soldar. Respeitando a projeção, estes são pingados no lugar por um operador. Após a pingagem de alguns componentes, dá-se início à soldadura automática dos componentes à viga, utilizando o robô.

3.2 Componentes e *Layout* da célula

A Tabela 12 apresenta todos os componentes base e principais da célula. Estes foram os componentes que, no início deste trabalho, foram considerados e a partir dos quais foi construída e desenvolvida toda a célula.

A figura seguinte, Figura 29, mostra o *layout* que foi pensado inicialmente. A disposição inicial dos componentes foi feita tendo como únicos requisitos o espaço limitado destinado para implementação da célula e o cumprimento dos objetivos principais do projeto, isto é, permitir o processo de soldadura colaborativa descrito no início deste capítulo.

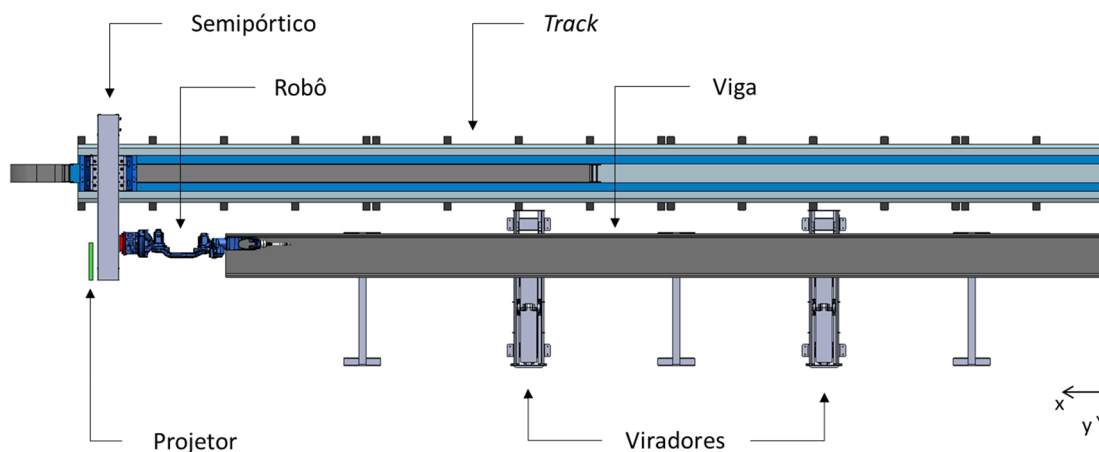


Figura 29 - Célula Robótica: *Layout* inicial.

Tabela 12 Célula robótica: Componentes base

Componentes	Descrição
Quadro elétrico	<p>Quadro elétrico geral de toda a célula, contendo:</p> <ul style="list-style-type: none"> • Fontes de alimentação DC • <i>Switch</i> de rede • Componentes do sistema de segurança • Módulos de IO's • Outros circuitos
Robô	YASKAWA Motoman MA1440 [35]
Controlador Robô	Motoman DX200 [36]
Semipórtico	Estrutura para o posicionamento do robô. Será fixo ao eixo linear
Eixo linear (<i>track</i>)	Eixo linear YASKAWA TSL [37], considerado eixo externo do robô.
HMI	Pc industrial e ecrã <i>touch</i> para o <i>software</i> da SARKKIS Robotics e <i>soft</i> PLC CodeSys
Máquina de soldar	<p>Fronius TransSteel 5000 Robotics [38]</p> <p>Máquina de soldar para robôs de soldadura. Inclui a máquina de soldar, com capacidade de comunicação por DeviceNet, e o Wire Feeder para montagem a bordo do robô</p>
Máquina de soldar (pingar)	Máquina de soldar compacta genérica com a função de pingar os componentes à viga
Sistema de deteção e guiamento de junta	<p>MOTOMAN ComArc [39]</p> <p>Sistema de deteção e guiamento de junta, a acoplar ao sistema de soldadura do robô, que utiliza informações relativas à corrente de soldadura</p>
Viradores	<p>Beam Rotator – Machinery Services</p> <p>Sistema hidráulico para rotação e movimentação de vigas</p>
Projetor	<p>MediaLas ILP 622 [40]</p> <p>Projetor laser para sinalizar o local dos componentes a pingar na viga</p>

3.3 Arquitetura do sistema

De modo a compreender o funcionamento de todos os componentes e a maneira como interagem/comunicam.

A Figura 30 detalha a arquitetura da totalidade do sistema representando todos os fluxos de informação. Contudo, esta figura refere-se à célula numa fase terminal do projeto.

Inicialmente, este esquema começou por ser muito simples, interligando apenas os componentes principais da célula. No entanto, ao longo deste trabalho foi sendo atualizado à medida que foram sendo tomadas decisões e acrescentadas funcionalidades, contendo cada vez mais informação e, consecutivamente, tornando-se útil para um maior número de pessoas.

Este esquema é de elevada importância para a compreensão da célula e para a organização conceptual de todos os componentes, pois agrega toda a informação relativa à célula e, dada a sua importância, foi recursivamente consultado pela equipa da SARKKIS Robotics e até mesmo por alguns fornecedores.

A construção deste esquema constitui um trabalho em curso, que só terminará quando a célula robótica estiver totalmente concluída.

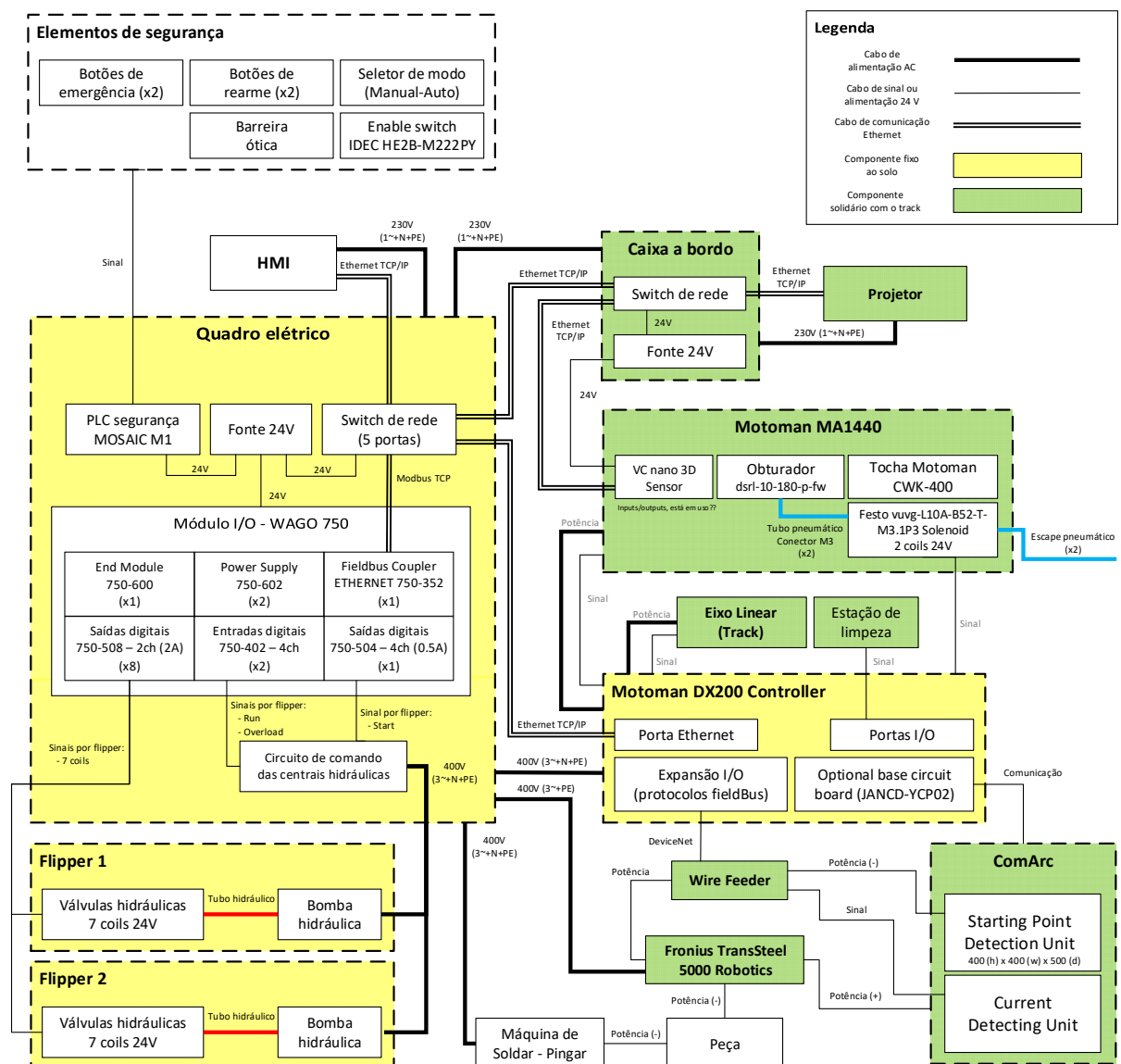


Figura 30 - Célula robótica: Fluxo de informação e Arquitetura do sistema.

4 Segurança da célula robótica

Do ponto de vista funcional uma máquina é tanto melhor quão melhor desempenhar a sua função, contudo, para poder ser utilizada deve ser segura. A segurança da máquina deve ser priorizada.

Neste capítulo é feita a implementação da solução para o sistema de segurança da célula. Inicia-se assim pelo exercício do *Risk Assessment* que constitui a base desta solução. É também detalhada a programação do PLC de segurança responsável pela monitorização e controlo do estado da célula.

4.1 Conceção da célula robótica

Este capítulo retrata todo o processo de avaliação e redução dos riscos realizado na conceção da célula robótica, incluindo, portanto, a definição do *layout* e das características de segurança. Para este exercício foi apenas considerada a fase de operação da máquina, ou seja, de todo o ciclo de vida da máquina, foca-se apenas na compreensão dos processos operacionais e das diferentes tarefas que lhe estão associadas.

Assim, com base neste exercício é construída a célula robótica, justificando a utilização e o posicionamento de cada elemento, quer seja ou não de segurança.

De modo a tornar o exercício da construção de uma célula robótica um processo recursivo e sistemático, foi criado um documento *template*. Este documento encontra-se disponível no Anexo A e foi contruído tendo como base a norma ISO/TR 14121-2 [8] e o estado da arte apresentado.

4.1.1 Zonas da célula

A Figura 31 apresenta as diferentes zonas da célula.

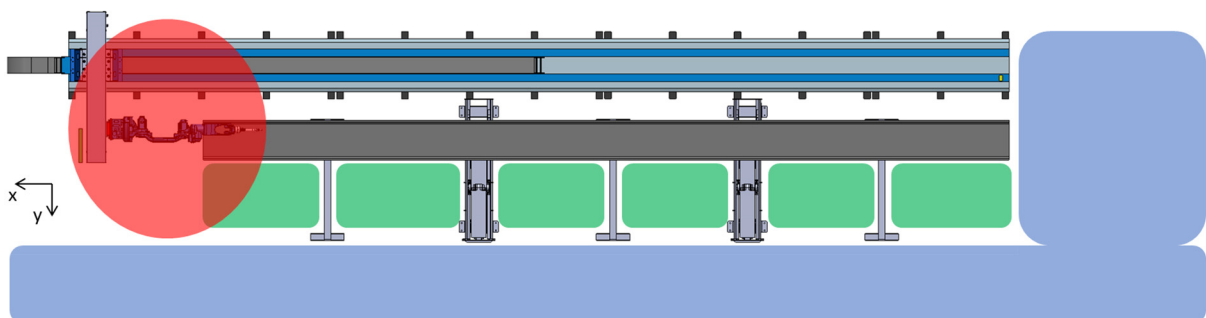


Figura 31 - Zonas da célula robótica: Vermelho-Zona de trabalho do robô; Verde-Zona de trabalho do operador; Azul-Vias de circulação.

A zona de trabalho do robô é delimitada pela zona a vermelho, esta acompanha o robô na sua movimentação ao longo do *track*.

A zona verde é a zona onde o operador trabalha, onde se realizam as operações de pingagem.

A zona azul é a zona reservada para movimentação do operador. A viga entra na célula da direita para a esquerda, direção e sentido do eixo X, e, é ao longo desta zona que o operador acompanha a sua entrada e saída.

4.1.2 Procedimento operacional

Neste subcapítulo são detalhadas todas as etapas elementares que constituem o procedimento de operação da máquina.

Carregamento

A célula deve estar totalmente parada.

A viga é transportada por uma ponte rolante.

O robô e os Viradores têm de estar na posição *home*, de modo a que a viga possa ser descarregada da ponte rolante.

O operador acompanha a operação de transporte e posicionamento da viga nos cavaletes, ficando a viga disposta paralelamente ao *track* do robô.

O abastecimento do carrinho auxiliar com todas as peças necessárias deve ser feito neste momento.

Calibração

O operador inicia, através da HMI, a calibração do sistema.

Pingagem

Após a calibração do sistema, têm início os processos de projeção e de pingagem de componentes à viga. O projetor é posicionado na 1ª zona de projeção, onde é feita a projeção.

É feito, pelo operador, o posicionamento e consecutiva pingagem das peças na posição correta, segundo a projeção.

Mudança de zona de projeção

No carrinho existem botões que habilitam o deslocamento do conjunto projetor-robô entre as várias zonas ou posições de projeção, permitindo continuar a projetar e pingar peças ao longo da viga.

Pingagem

Estando todas as peças a soldar pingadas na sua posição, a etapa de preparação está terminada e o operador deve dirigir-se à HMI.

Após pingagem de todos os componentes desta zona de projeção, o operador deve dirigir-se ao carrinho auxiliar.

Soldadura

No HMI, o operador dá início à soldadura de todos os componentes.

No final da operação o robô deve ficar “arrumado” na sua posição *home*.

Nota:

Existem duas possibilidades:

- pinga-se e solda-se duas faces, roda-se a viga, pinga-se e solda-se as duas faces restantes. Nesta opção a viga é rodada uma vez.
- pinga-se as quatro faces e só depois se solda. Neste caso, fazem-se duas rotações à viga, uma durante a pingagem e outra durante a soldadura.

Controlo de qualidade

Finalizando o processo de soldadura, o operador procede ao controlo de qualidade.

Descarregamento

Durante esta etapa, a célula deve estar totalmente parada.

O operador fixa a viga à ponte rolante e a viga é transportada para o exterior da célula.

4.1.3 Limites da máquina

Tendo em mente o esquema da Figura 2, a definição dos limites da máquina é a primeira etapa da avaliação dos riscos. De seguida, são definidos os limites de usabilidade e espaciais.

Limites de usabilidade

Os limites de usabilidade podem ser divididos em duas partes: a utilização prevista e a má utilização razoavelmente previsível.

Utilização prevista

A máquina é concebida para soldadura em vigas de aço de construção *standard*.

As dimensões máximas e mínimas para o comprimento da viga são, respetivamente, 12m e 4m. Em termos de largura da secção, esta deve estar compreendida entre 80 e 600mm.

A viga deve ser posicionada em cima de cavaletes, dentro da zona de ação do robô.

A célula está concebida para ser operada por apenas um operador. Este deve ser capaz de realizar processos de pingagem e conhecer o funcionamento HMI.

Má utilização razoavelmente previsível

O mau posicionamento inicial da viga pode resultar na impossibilidade do sistema calibrar a posição da viga.

Entrada de pessoas estranhas no espaço reservado à máquina, uma vez que este espaço se encontra numa zona movimentada da fábrica.

Limites espaciais

A célula será instalada em ambiente industrial coberto.

Necessidade de uma fonte de energia elétrica, 400V trifásica.

Necessidade de uma tomada de ar comprimido.

Necessidade de uma tomada de gás de soldadura.

Tratando-se de um processo de soldadura, o espaço onde a célula for instalada deve ser ventilado para extração de fumos.

A área máxima para implementação da célula é, no máximo, um retângulo de 7×18m.

O operador deverá ter espaço suficiente entre os cavaletes para poder pingar os componentes, de forma confortável, a todo o comprimento da viga.

4.1.4 Identificação dos perigos

Na tabela seguinte, Tabela 13, é feita a identificação dos perigos, identificando também, para cada perigo, a tarefa ou operação em que ele aparece, a situação potencialmente perigosa e o evento perigoso que pode ser despoletado.

Tabela 13 - Identificação dos perigos

Identificação dos perigos					
Máquina (identificação)		Célula CoopWeld		Analista	
Referências		Especificações, design da célula		Versão	1.1
Enquadramento (fase do ciclo de vida; parte/função da máquina)		Operação		Data	20/02/2017
Metodologia		ISO 10218-2:2011, Anexo A ISO 14121-1:2007, Anexo A ISO 12100			
Nº Ref.	Zona de perigo	Tarefa/Operação	Perigo	Situação perigosa	Evento perigoso
1	Zona de trabalho do robô	<ul style="list-style-type: none"> • Calibração • Mudança de zona de projeção • Soldadura 	<ul style="list-style-type: none"> • Impacto, a diferentes velocidades, forças e inércias • Corte (tocha de soldadura) • Fricção/abrasão • Esmagamento 	Presença do operador na zona de trabalho do robô	Contacto imprevisto com o operador devido ao movimento dos seguintes componentes, isoladamente ou em conjunto: <ul style="list-style-type: none"> • Braço do robô • Tocha de soldadura • Semipórtico, por ação do eixo linear • Viradores
2	Zona de transporte de viga e zona de trabalho do operador	<ul style="list-style-type: none"> • Carregamento • Pingagem • Controlo de qualidade • Descarregamento 	<ul style="list-style-type: none"> • Corte • Fricção/abrasão 	Presença de componentes com arestas vivas ou pontas afiadas na área de trabalho do operador Ex: Ponta da tocha de soldadura	Contacto com componentes e/ou ferramentas, com arestas vivas ou pontas afiadas, devido a descuido ou falta de atenção do operador

3	Zona de transporte de viga	<ul style="list-style-type: none"> • Carregamento • Descarregamento 	<ul style="list-style-type: none"> • Impacto a baixa velocidade e elevada inércia • Esmagamento • Fricção/abrasão 	Guiamento manual durante o transporte e posicionamento da viga na zona de trabalho do robô	Descuido ou falta de atenção do operador durante o manuseamento de peças de grandes dimensões e elevada massa
4	Zona de transporte de viga e zona de trabalho do operador	<ul style="list-style-type: none"> • Pingagem • Soldadura • Controlo de qualidade • Descarregamento 	Queimadura	Existência de componentes ou materiais a elevadas temperaturas na zona de trabalho do operador	Contacto com superfícies quentes Exemplo: <ul style="list-style-type: none"> • Tocha de soldadura • Peça acabada de soldar/pingar
5	Zona de trabalho do robô	<ul style="list-style-type: none"> • Pingagem • Soldadura 	Ferimento ocular e/ou pele	Exposição do operador à radiação do arco elétrico	Emissão de radiação resultante do processo de soldadura
6	Zona de trabalho do operador	<ul style="list-style-type: none"> • Pingagem 	Ferimento ocular	Exposição do operador à projeção do laser	Emissão de radiação proveniente do projetor laser
7	Zona de trabalho do operador	<ul style="list-style-type: none"> • Pingagem • Soldadura 	Fumos	Exposição do operador a fumos de soldadura	Emissão de fumos resultantes do processo de soldadura/pingagem
8	Zona de trabalho do operador	<ul style="list-style-type: none"> • Pingagem • Soldadura 	<ul style="list-style-type: none"> • Queimadura • Ferimento ocular e/ou pele 	Exposição do operador ao processo de soldadura	Salpico de partículas incandescentes, resultantes do processo de soldadura
9	Zona de trabalho do operador	<ul style="list-style-type: none"> • Carregamento • Pingagem • Soldadura • Controlo de qualidade • Descarregamento 	Elétrico	Manuseamento dos diversos equipamentos	Contacto com partes ativas da máquina
10	Zona de trabalho do robô	<ul style="list-style-type: none"> • Calibração • Pingagem • Mudança de zona de projeção • Soldadura 	Impacto por ejeção de ferramentas, peças, componentes ou outro equipamento	Acondicionamento inadequado dos equipamentos auxiliares ao processo. Existência de componentes nas vias de circulação.	Contacto com o robô ou outro componente robotizado móvel com: <ul style="list-style-type: none"> • Carrinho • Máquina de soldar (pingagem) não arrumada • Outra peça ou ferramenta utilizada pelo operador Ejeção de elementos soltos

4.1.5 Estimativa, Valoração e Redução dos riscos

A tabela seguinte, Tabela 14, apresenta a estimativa dos riscos e a sua valoração, apresentando medidas para a redução dos riscos e a valoração dos riscos residuais.

Tabela 14 - Estimativa, Valoração e Redução dos riscos

Nº Ref.	Estimativa do risco				Redução do risco	Estimativa do risco residual				Valoração do risco
	S	F	P	PLr	Medidas de proteção	S	F	P	PLr	
1	2	2	1	d	Instalação de barreiras de segurança em torno da célula. Implementar um sistema de controlo garantindo que durante as tarefas referidas o operador não está no interior do espaço protegido.	1	1	1	a	Acontece nas tarefas onde existe movimentação dos componentes robotizados. A utilização de barreiras físicas comparativamente a barreiras luminosas é preferível. As barreiras físicas têm um preço mais baixo e podem ser instaladas a uma menor distância do ponto perigoso, comparativamente às barreiras luminosas. Risco residual tolerável.
2	2	1	1	c	O operador deve estar familiarizado com o ambiente industrial direcionado para a soldadura.	-	-	-	-	Risco tolerável.
3	1	1	1	a	O operador deve estar familiarizado com o ambiente industrial e com o manuseamento de cargas.	-	-	-	-	Risco tolerável.
4	1	2	1	b	O operador deve estar familiarizado com o processo de soldadura. Deve utilizar o equipamento de segurança adequado (máscara, luvas e vestuário adequado).	-	-	-	-	Risco tolerável.
5	1	2	1	b	Medidas de proteção semelhantes à Ref. 4.	-	-	-	-	Implicações na saúde, a longo prazo. Risco tolerável.
6	1	2	1	b	O operador deve ser informado acerca dos perigos relativos aos equipamentos presentes na célula.	-	-	-	-	Implicações na saúde, a longo prazo. Risco tolerável.
7	1	2	2	c	Medidas de proteção semelhantes à Ref. 4.	-	-	-	-	Risco tolerável.

					A célula deve ser instalada num local ventilado.					
8	1	2	1	b	Medidas de proteção semelhantes à Ref. 4.	-	-	-	-	Risco tolerável.
9	1	1	2	b	Realização da manutenção periódica dos equipamentos, atempadamente.	-	-	-	-	Risco tolerável.
10	2	2	1	d	Centralizar todo o equipamento necessário para o processo de pingagem numa bancada móvel (máquina de soldar e outras ferramentas). Esta bancada deverá ser munida de um sistema de guiamento, que apenas permite o seu movimento paralelamente ao <i>track</i> , mantendo uma distância constante entre a bancada e o <i>track</i> , impossibilitando, assim, a sua entrada na zona de operação do robô e que esta seja deixada nas vias de circulação.	2	1	1	c	O sistema de guiamento poderá ser fixo a uma das barreiras mencionadas em 1 (Nº ref 1). Risco residual tolerável.

Para perigos de Ref. nº1 e nº10, é necessário aplicar medidas de redução dos riscos substanciais. Assim, optou-se por construir um gradeamento em torno da célula para prevenir a entrada de pessoas durante operações e tarefas críticas, onde ninguém pode estar no seu interior. Além disso, para garantir a segurança do operador, enquanto este se encontra dentro da célula, a bancada tem um sistema de guiamento que garante que esta se encontra sempre numa zona segura, fora do alcance dos componentes móveis e fora das vias de circulação do operador.

Barreiras de segurança

Nas medidas de redução de risco acima descritas, concluiu-se que a implementação de barreiras em torno da célula é necessária para a redução do perigo, estando este identificado como de elevado risco aquando do funcionamento automático (tarefa de soldadura).

Surgiram três possibilidades para a implementação destas barreiras, tais como:

- Utilizar apenas barreiras luminosas.
- Utilizar apenas barreiras físicas (gradeamento) com um portão automático nas zonas de entrada e saída de pessoas e ou materiais.
- Utilizar ambos os tipos de barreiras.

Como as barreiras luminosas são consideravelmente mais caras que as físicas, é dada preferência à utilização de barreiras físicas, sendo a segunda opção a mais económica.

No entanto, e considerando ainda a segunda hipótese, a utilização de uma ou mais barreiras luminosas no lugar do(s) portão(ões) automático(s) traz vantagens quer ao nível da facilidade de utilização, quer no tempo necessário para se entrar e sair da célula.

Assim, ficou decidido, em conjunto com a equipa da SARKKIS Robotics, que se utilizariam barreiras físicas em torno de toda a máquina, exceto nas zonas de acesso, onde se utilizariam barreiras luminosas.

Identificação dos elementos a tratar

A zona de acesso fica à direita da máquina. Assim, entrada das vigas através da ponte rolante é facilitada, permitindo deslocar a viga a uma altura menor e permite, ainda, que o operador acompanhe o seu movimento, uma vez que não há nenhum obstáculo no seu caminho até aos Viradores.

A Figura 32 identifica as barreiras que foram seleccionadas e ilustra o seu posicionamento relativamente à máquina.

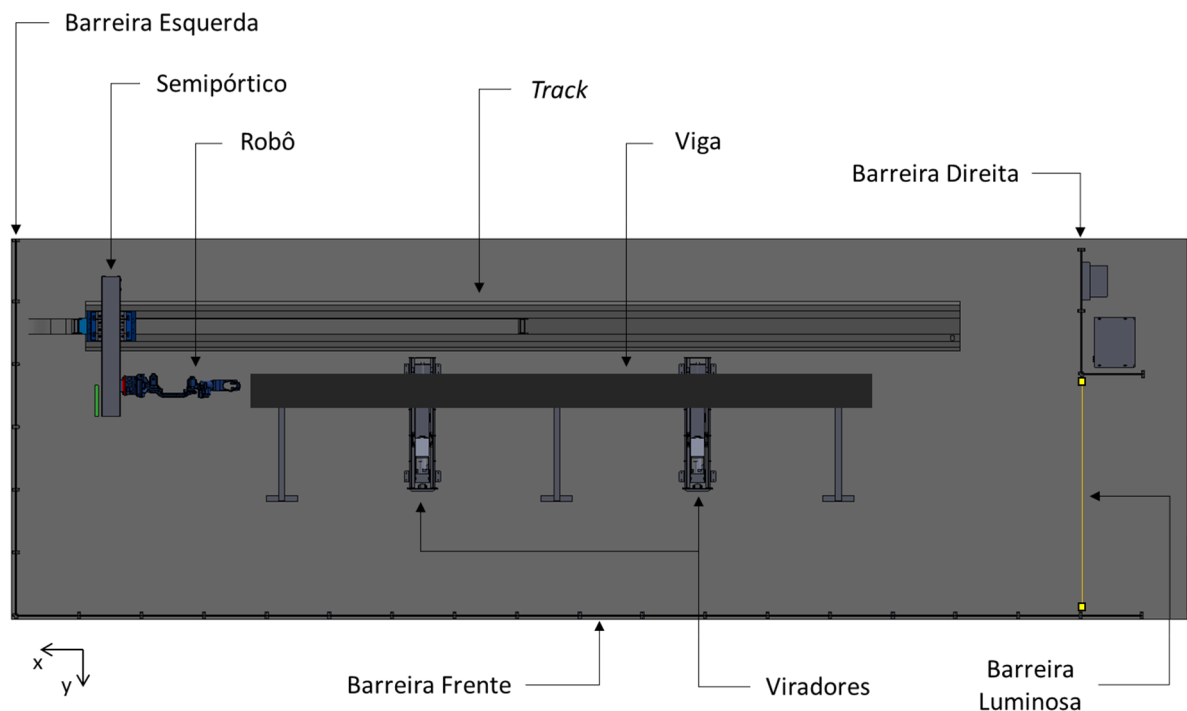


Figura 32 - Identificação das barreiras de segurança (vista de topo).

- Barreira Esquerda – barreira física perpendicular ao *track*, mais à esquerda da célula
- Barreira Frente – barreira física paralela ao *track*
- Barreira Direita – barreira física perpendicular ao *track*, mais à direita da célula
- Barreira Luminosa – barreira luminosa perpendicular ao *track* na zona de acesso à máquina

De seguida, é feito o levantamento dos pontos potencialmente perigosos relacionados a cada barreira. Assim, é possível especificar a distância mínima de segurança entre o ponto potencialmente perigoso e a barreira em causa.

Barreira Esquerda

A Figura 33 ilustra os pontos potencialmente perigosos relativos à Barreira Esquerda. O robô está posicionado numa posição limite (dimensões em *mm*).

- Ponto A – projetor
- Ponto B – tocha de soldadura

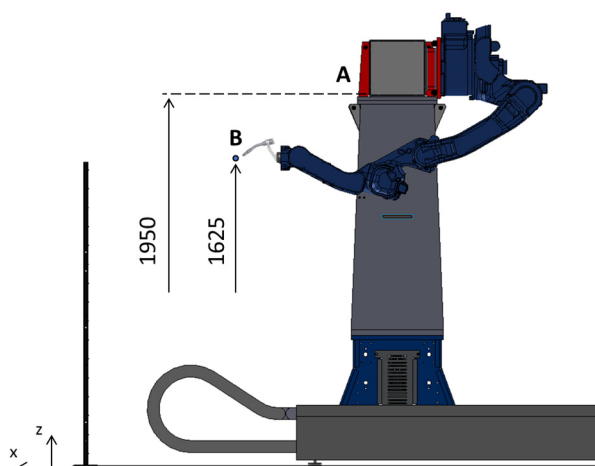


Figura 33 - Pontos potencialmente perigosos da Barreira Esquerda.

Tendo em conta que, quanto mais baixas forem as barreiras, mais económicas se tornam. Consideram-se ideais, para esta aplicação, as barreiras físicas com alturas compreendidas entre os 1400 e os 1800mm.

Para situações de alto risco, a Tabela 15 identifica as distâncias para os pontos anteriormente referidos, Pontos A e B.

Para as alturas de barreira pretendidas, as distâncias mínimas entre a barreira e o ponto perigoso são as mesmas, independentemente do ponto considerado. O ponto B é o ponto mais próximo da barreira, logo a distância de segurança deve ser medida partindo deste ponto.

Tabela 15 - Distâncias mínimas de segurança para as Barreira Esquerda, Barreira Direita e Barreira Frente

Dimension in millimetres

Height of hazard zone <i>a</i>	Height of protective structure <i>b</i>									
	1 000	1 200	1 400	1 600	1 800	2 000	2 200	2 400	2 500	2 700
	Horizontal safety distance to hazard zone, <i>c</i>									
2 700	0	0	0	0	0	0	0	0	0	0
2 600	900	800	700	600	600	500	400	300	100	0
2 400	1 100	1 000	900	800	700	600	400	300	100	0
2 200	1 300	1 200	A E 1 000	900	800	600	400	300	0	0
2 000	1 400	1 300	C 1 100	900	800	600	400	0	0	0
1 800	1 500	1 400	1 100	900	800	600	0	0	0	0
1 600	1 500	1 400	B 1 100	900	800	500	0	0	0	0
1 400	1 500	1 400	1 100	900	800	0	0	0	0	0
1 200	1 500	1 400	1 100	900	700	0	0	0	0	0
1 000	1 500	1 400	1 000	800	0	0	0	0	0	0
800	1 500	1 300	D 900	600	0	0	0	0	0	0
600	1 400	1 300	800	0	0	0	0	0	0	0
400	1 400	1 200	400	0	0	0	0	0	0	0
200	1 200	900	0	0	0	0	0	0	0	0
0	1 100	500	0	0	0	0	0	0	0	0

A escolha recai para uma barreira com altura de 1600mm , sendo esta a opção com altura mais baixa sem comprometer em demasia a distância mínima de segurança ao ponto perigoso.

Conclui-se assim, que a distância mínima a respeitar, de acordo com a norma ISO 13857 [14], é de 900mm , como ilustrado na Figura 34 (dimensões em mm).

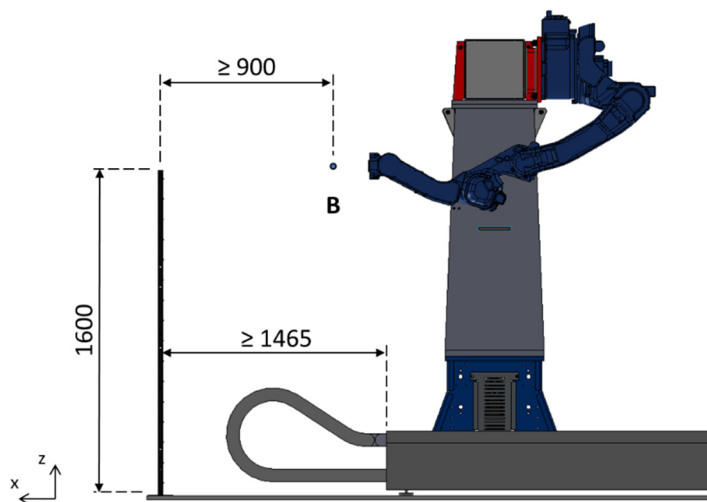


Figura 34 - Distância mínima de segurança para a Barreira Esquerda.

Barreira Frente

A Figura 35 ilustra os pontos potencialmente perigosos relativos à Barreira Frente. O robô está posicionado na posição de maior alcance em direção a esta barreira (dimensões em mm).

- Ponto C – tocha de soldadura
- Ponto D – parte móvel do Virador

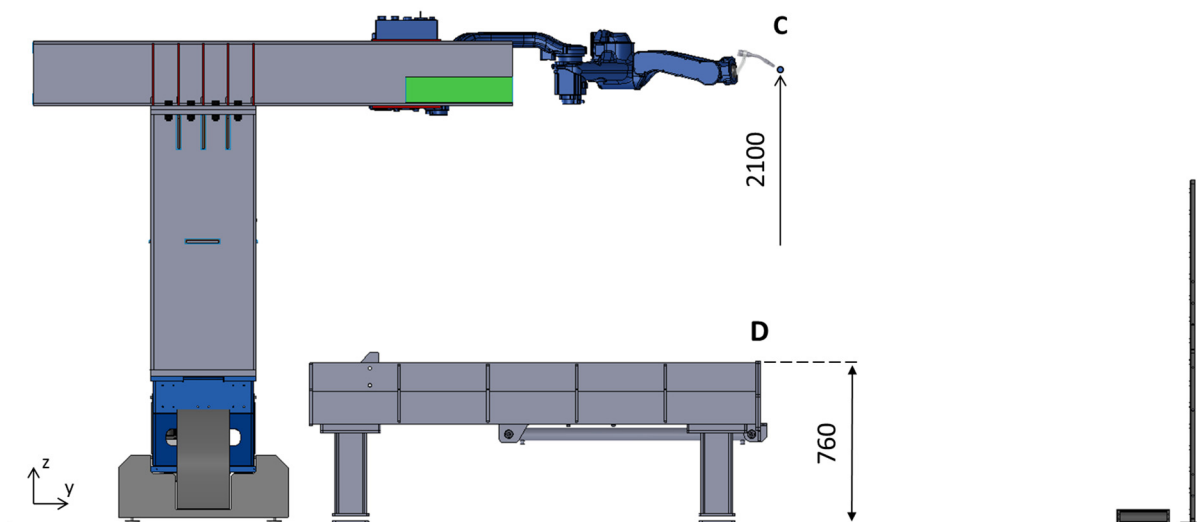


Figura 35 - Pontos potencialmente perigosos da Barreira Frente.

Seguindo o raciocínio anterior, relativo à altura das barreiras de segurança, e de modo a uniformizar as alturas de todas as barreiras a utilizar, a utilização de barreiras com 1600mm de altura é preferível.

A Tabela 15 acima apresentada identifica, para situações de risco elevado, as distâncias para os Pontos C e D.

Retira-se, assim, que para este caso, o Ponto C é o ponto perigoso a ser considerado e a partir do qual se deve aplicar uma distância de segurança no mínimo de 900mm. A figura seguinte, Figura 36, ilustra a distância mínima a respeitar (dimensões em mm), de acordo com a norma ISO 13857 [14].

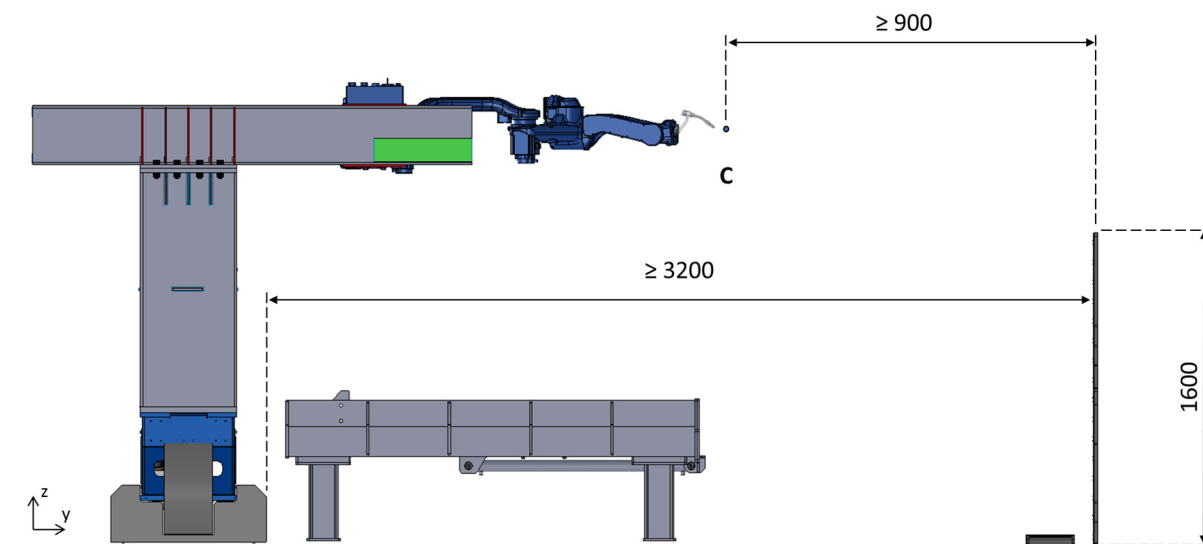


Figura 36 - Distância mínima de segurança para a Barreira Frente.

Barreira Direita

Relativamente à Barreira Direita, o Ponto E é identificado como o ponto perigoso mais próximo da barreira, estando o robô posicionado numa posição limite. A Figura 37 ilustra esta situação (dimensões em mm).

- Ponto E – tocha de soldadura

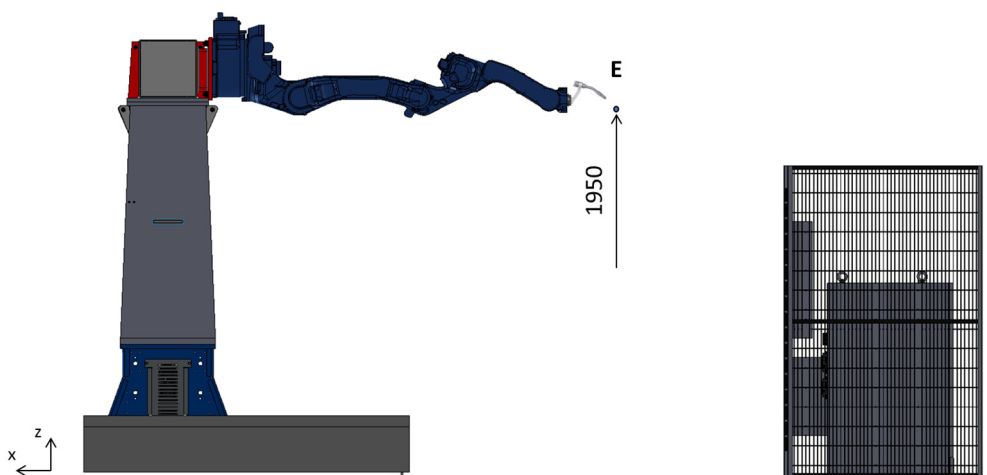


Figura 37 - Pontos potencialmente perigosos da Barreira Direita.

Mais uma vez, a Tabela 15 ilustra as distâncias mínimas de segurança relativas ao Ponto E.

A escolha da barreira é direcionada, mais uma vez, para barreiras com altura de $1600mm$. A distância mínima a respeitar, de acordo com a norma ISO 13857 [14], é de $900mm$. A figura seguinte, Figura 38, demonstra esta distância (dimensões em mm).

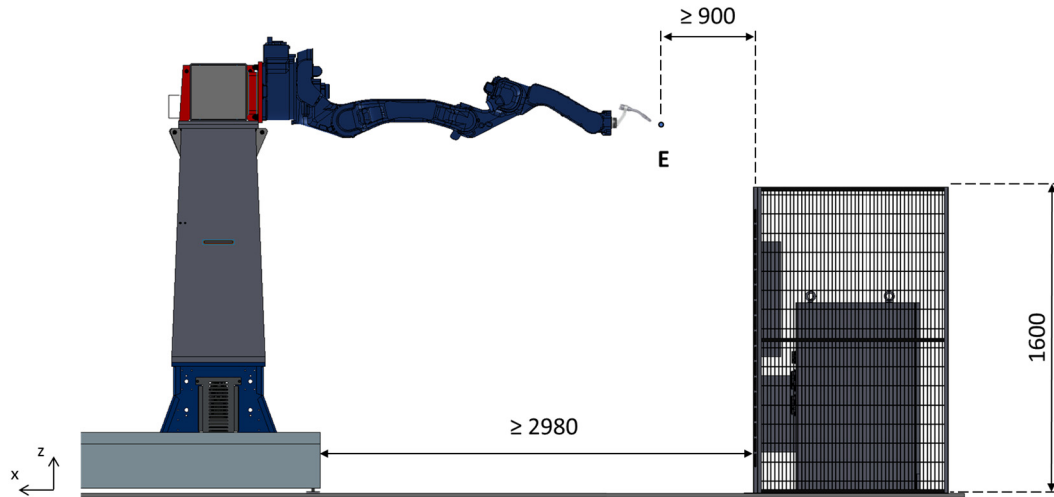


Figura 38 - Distância mínima de segurança para a Barreira Direita.

Barreira Luminosa

A Barreira Luminosa encontra-se na mesma situação que a Barreira Direita.

Após o debate com a equipa da SARKKIS Robotics acerca do tempo total de paragem dos equipamentos, T , estima-se, numa primeira fase, que este seja de:

$$T = 0,425s [41]$$

O tempo de paragem do *track* não foi contabilizado, pois considera-se que este se encontra no final do seu curso, quer seja um fim de curso físico ou virtual (*software*).

O valor do tempo de paragem total, T , deverá, à posteriori ser confirmado experimentalmente e, se necessário, refeitos os cálculos que serão apresentados de seguida.

O cálculo da distância mínima de segurança deve considerar duas situações para a forma como um indivíduo cruza a barreira luminosa em direção ao perigo.

- **O indivíduo dirige-se ao perigo, por cima da barreira, numa tentativa de a contornar.**

Tal como está descrito no estado da arte para esta situação deve utilizar-se a equação 12:

$$S = (2000 \times T) + C_{RO}$$

A altura do feixe superior da barreira será de $1600mm$ de modo a ficar ao mesmo nível das barreiras físicas.

A

Tabela 16 demonstra o valor de C_{RO} que se aplica nesta situação.

Tabela 16 - Configuração nº1: Valor de C_{RO} para a Barreira Luminosa

Dimensions in millimetres

Height of hazard zone <i>a</i>	Height of upper edge of the detection zone of the electro-sensitive protective equipment <i>b</i>											
	900	1 000	1 100	1 200	1 300	1 400	1 600	1 800	2 000	2 200	2 400	2 600
	Additional distance to hazard zone C_{RO}											
2 600 ^a	0	0	0	0	0	0	0	0	0	0	0	0
2 500	400	400	350	300	300	300	300	300	250	150	100	0
2 400	550	550	550	500	450	450	400	400	300	250	100	0
2 200	800	750	750	700	650	650	600	550	400	250	0	0
2 000	950	950	850	850	800	750	700	550	400	0	0	0
1 800	1 100	1 100	950	950	850	800	750	550	0	0	0	0
1 600	1 150	1 150	1 100	1 000	900	850	750	450	0	0	0	0
1 400	1 200	1 200	1 100	1 000	900	850	650	0	0	0	0	0
1 200	1 200	1 200	1 100	1 000	850	800	0	0	0	0	0	0
1 000	1 200	1 150	1 050	950	750	700	0	0	0	0	0	0
800	1 150	1 050	950	800	500	450	0	0	0	0	0	0
600	1 050	950	750	550	0	0	0	0	0	0	0	0
400	900	700	0	0	0	0	0	0	0	0	0	0
200	600	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0

Considerando o pior cenário possível e uma vez que, para a mesma altura do feixe superior, a distância adicional varia pouco com a altura do perigo, considera-se 750mm como a distância adicional C_{RO} a aplicar.

Assim,

$$S = 1600mm$$

Contudo, sendo este valor é superior a 500mm, deve utilizar-se a equação 13:

$$S = (1600 \times T) + C_{RO}$$

Obtendo-se uma distância mínima de segurança de:

$$S = 1430mm$$

- A direção de aproximação do indivíduo é ortogonal à zona de detecção da barreira.

De modo a que a distância de segurança, nesta situação, seja inferior à da situação anterior, deverá selecionar-se uma barreira com $d \leq 40mm$. Assim, o valor da distância mínima de segurança pode ser calculado através da equação 6:

$$S = (1600 \times T) + 8(d - 14)$$

Obtendo-se:

$$S = 888mm$$

Conclui-se assim que a distância a respeitar será de 1430mm, como ilustra a Figura 39 (dimensões em mm):

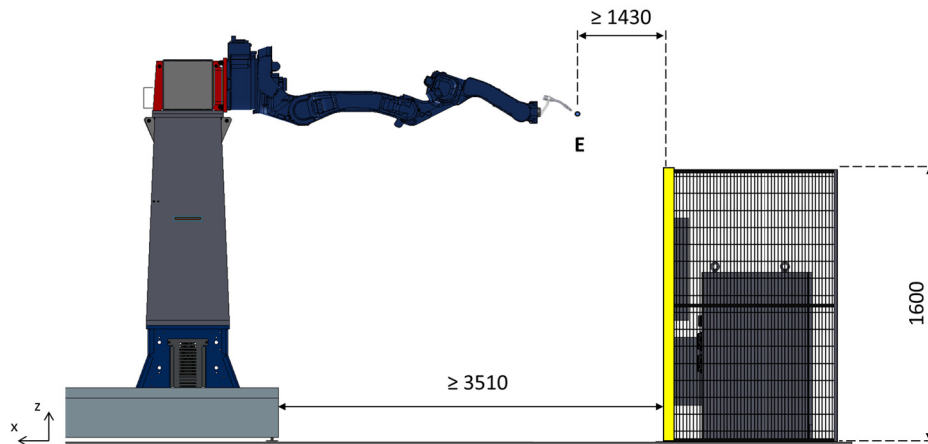


Figura 39 - Configuração nº1: Distância mínima de segurança para a Barreira Luminosa

Escolha da barreira luminosa

A barreira luminosa a utilizar tem os seguintes requisitos:

- Nível de performance PLd, de acordo com a Ref. nº1 da Tabela 14
- Feixe superior a 1600mm de altura e feixe inferior a uma altura máxima de 300mm, o que corresponde a uma altura protegida de no mínimo 1300mm
- Capacidade de deteção $d \leq 40mm$

Assim, dando preferência ao equipamento do mesmo fornecedor do PLC de segurança (ReeR [42], de acordo com o PLC de segurança escolhido no capítulo 4.2.1) foi escolhida a barreira EOS4 A 40 mm *resolution model* 1354, ilustrada pela Figura 40.



Figura 40 - Barreira luminosa ReeR EOS4 A 40 mm *resolution model* 1354 [42].

Validação das distâncias mínimas de segurança das barreiras

O exercício de dimensionamento de distâncias de segurança para barreiras físicas foi realizado idealizando a célula a implementar sem qualquer tipo de limitação espacial. Na realidade, o espaço visado para a implementação da célula, cedido pela Norfer, é relativamente pequeno, sendo impossível colocar as barreiras às distâncias acima especificadas.

A solução passa por limitar a área de trabalho do robô e limitar o curso do *track*.

Feita uma análise experimental em ambiente de simulação CAD, conclui-se que a limitação da área de trabalho do robô segundo o eixo X , $490mm$ em X^- e $1715mm$ em X^+ (valores relativos à base do robô), garante um bom compromisso entre a usabilidade do robô durante o processo e o espaço disponível para implementação da célula.

A Figura 41 seguinte ilustra a área de trabalho limitada do robô (dimensões em mm).

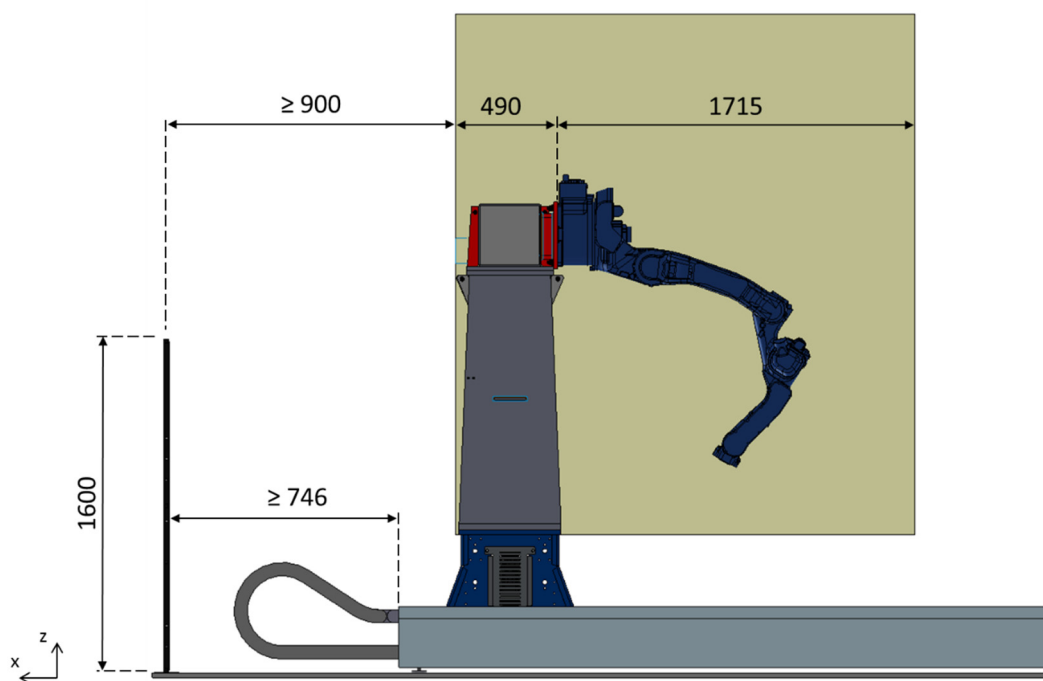


Figura 41 - Configuração n°1: Limites da área de trabalho segundo o eixo X considerando o semipórtico posicionado o mais próximo da Barreira Esquerda.

Nesta figura, o pórtico encontra-se na posição mais próxima possível da Barreira Esquerda. Com a limitação de $490mm$ em X^- e, respeitando a distância mínima de $900mm$ à barreira, conseguem-se $746mm$ de distância entre a extremidade do *track* e a barreira. Esta distância é suficiente para o bom funcionamento do *track* e facilita a passagem de uma pessoa caso seja necessário (manutenção).

A limitação de $1715mm$ em X^+ foi estabelecida em conjunto com a equipa da SARKKIS Robotics e permite que o robô desempenhe todos os movimentos que possa vir a descrever. Este valor foi conseguido dando uma margem de movimento de $350mm$ além da junta 3 na direção X .

Contudo, a limitação do movimento do robô relativamente à sua base não é suficiente para cumprir os requisitos apertados espaciais, relativos à parte direita da célula. A solução passa por limitar, ao nível do *software* e *hardware*, o movimento do *track*.

Recorrendo, novamente, ao ambiente de simulação CAD, conclui-se que se deve limitar em 1330mm a posição extrema do *track*, garantindo, assim, a distância de segurança de 1430mm imposta anteriormente.

A figura seguinte, Figura 42, ilustra esta situação (dimensões em *mm*).

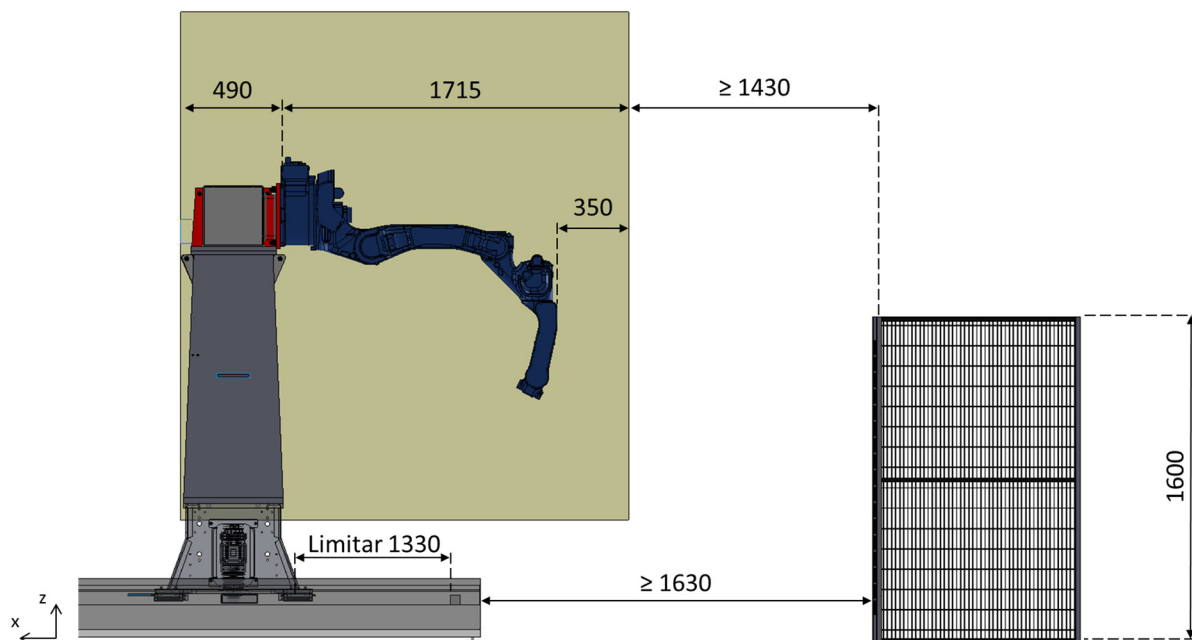


Figura 42 – Configuração n°1: limites da área de trabalho segundo o eixo Y considerando o pórtico posicionado o mais próximo da Barreira Direita.

Bancada

Do exercício de avaliação e redução dos riscos foi apresentada, como medida de redução do risco, a centralização de todo o equipamento necessário para o processo de pingagem numa bancada móvel (máquina de soldar e outras ferramentas) e, ainda, a implementação de um sistema de guiamento, que apenas permita o seu movimento paralelamente ao *track*, mantendo uma distância constante entre a bancada e o mesmo, impossibilitando, assim, a sua entrada na zona de operação do robô.

Limites e requisitos

De seguida, são apresentados alguns limites e requisitos estruturais para a bancada.

➤ Limite espacial

A imagem seguinte, Figura 43, apresenta uma análise do espaço disponível na célula para a bancada.

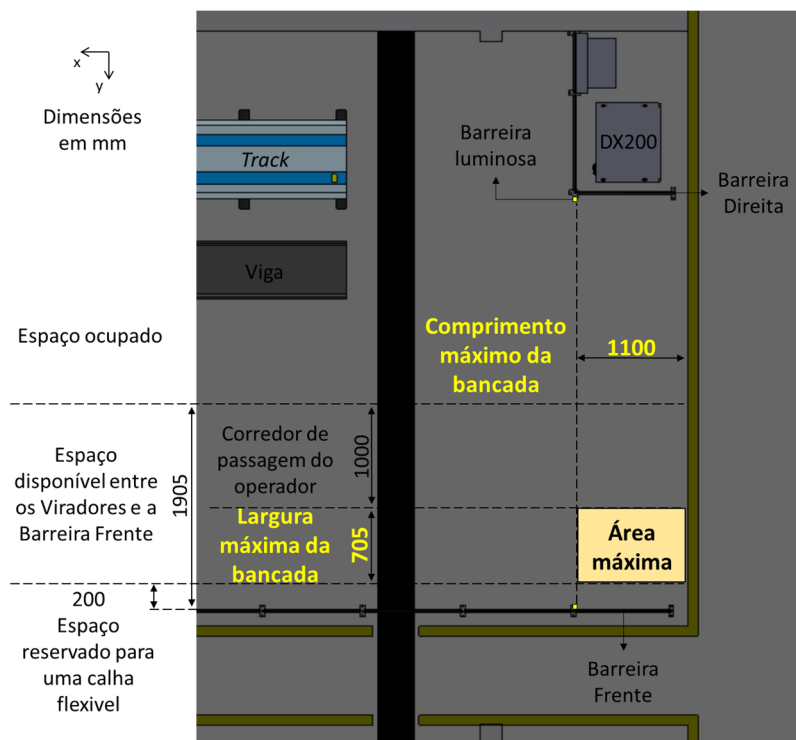


Figura 43 – Configuração nº1: Análise do espaço disponível para a bancada, vista de topo.

O espaço disponível em chão de fábrica é limitado e, de acordo com o procedimento operacional idealizado, a bancada deve ficar no exterior da área protegida para se dar início ao processo automático, sendo que o comprimento máximo disponível para a bancada é de 1100mm.

A largura máxima disponível para a bancada é de 705mm. Apesar da distância entre os Viradores e a barreira física (Barreira Frente) ser de 1905mm, é necessário reservar 1000mm para um corredor de passagem do operador e 200mm para a instalação de uma calha flexível, que transportará todos os cabos necessários para ligar os componentes a bordo da bancada e os restantes equipamentos.

➤ Peças a soldar

De seguida, foi desenvolvida pela SARKKIS Robotics uma estimativa das cargas envolvidas num projeto genérico de soldadura de uma viga de perfil HEA 300 com os seguintes componentes:

(consideram-se todas as dimensões em metros e densidade do aço de 7850 kg/m^3)

- Chapa de base $0,5 \times 0,4 \times 0,02 \times 7850 = 31,4 \text{ kg}$
- Chapa de topo $0,3 \times 0,29 \times 0,02 \times 7850 = 13,7 \text{ kg}$
- 16 Rigidificadores $16 \times 0,145 \times 0,26 \times 0,02 \times 7850 = 34,4 \text{ kg}$
- Total = 79,5kg

Desta forma, para um caso genérico comum, temos 79,5kg em componentes a serem soldados.

➤ Tampo da bancada

Sendo a bancada destinada ao transporte de componentes metálicos, é necessário que esta seja robusta e que aguente com sucessivas cargas e descargas de material.

Assim, o tampo da mesa será concebido a partir de uma chapa de aço. De modo a garantir a sua durabilidade, a chapa a utilizar deve ter uma espessura mínima de 5mm.

Fazendo uma análise de massa do tampo, considerando uma chapa com a dimensão máxima admissível: $0,705 \times 1,100$ (dimensões em metros), espessura de 5mm e densidade de $7850kg/m^3$ tem-se que a massa da chapa é de 30,4kg.

➤ Capacidade de carga

A tabela seguinte sumariza todos os equipamentos integrados na bancada e estima a sua massa, permitindo, assim, chegar a um valor para a capacidade de carga que a bancada deve apresentar.

Tabela 17 – Configuração nº1: Capacidade de carga

Componente	Massa [kg]
Tampo	30,4
Máquina de soldar (pingar)	50
Conjunto HMI	30
Componentes a soldar	79,5
Total:	189,9

Escolha da bancada

Para bancada móvel e em conjunto com a equipa da SARKKIS Robotics decidiu-se adquirir um carro de transporte industrial com rodas, que posteriormente será modificado para ir ao encontro dos objetivos do presente projeto.

Existem diversos fornecedores de carros de transporte industrial. Assim, foi realizada uma lista extensiva de todos os possíveis carros. Após discussão com a equipa, foi escolhido o carro da KAIZER+KRAFT [43]. A Tabela 18 apresenta as principais características deste produto.

Tabela 18 – Configuração nº1: Características carro de transporte KARIZER+KRAFT [43]


Marca	KAIZER+KRAFT	
Modelo	Eurokraft Active Green M1024006	
Capacidade de Carga [kg]	500	
Superfície útil [Larg. mm]×[Com. mm]	700 × 1050	
Altura plataforma sup. [mm]	895	
Diâmetro dos rodízios [mm]	200	
Garantia [anos]	10	
Preço [€]	267	

Figura 44 – Configuração nº1: Carro de transporte KARIZER+KRAFT [43].

Este carro cumpre todos os requisitos, à exceção do tampo uma vez que é em plástico. Assim, terá de se fabricar um tampo em aço à medida que respeite os requisitos mencionados na secção anterior, Tampo da bancada.

Carril de guiamento

O sistema de guiamento tem como função garantir que a bancada se encontra a uma distância constante do *track*, permitindo que ela se mova paralelamente a este.

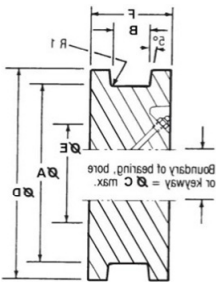
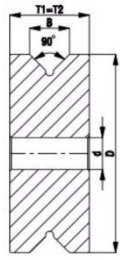
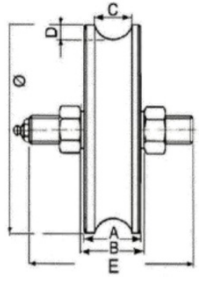
Para a conceção deste sistema de guiamento pretende-se minimizar as modificações à bancada base (carro de transporte) e conseguir um sistema robusto e de fácil instalação.

De seguida, na Tabela 19, é feita uma comparação entre os vários materiais para a composição das rodas. Na Tabela 20, são mostrados os principais tipos de carris disponíveis no mercado adequados para a presente aplicação.

Tabela 19 – Configuração nº1: Comparação entre o material do piso para rodas de rodízios fixos [44]

	Nylon	Aço/Ferro fundido	Importância
Ruído	Médio	Alto	Não
Vibração	Média	Alta	Sim (HMI)
Durabilidade	Alta	Muito alta	Sim
Carga	Alta	Muito alta	Sim
Resistência de rolamento	Baixa	Baixa	Sim

Tabela 20 – Configuração nº1: Rodas para diferentes tipos de carris [5]

Carril retangular	Carril em V	Carril em U
		

Foi elaborada uma lista com as várias opções possíveis e em conjunto com a equipa da SARKKIS Robotics foi escolhido, como melhor opção para o presente projeto, o conjunto da Blickle [5], ilustrado na Figura 45.



Figura 45 – Configuração nº1: Conjunto de roda para carril Blickle [5].

Este conjunto é constituído por um suporte e por uma roda em aço, para carril em V.

Duas das rodas presentes no carro de transporte serão substituídas por dois destes conjuntos. Como carril, é utilizada uma simples cantoneira em aço fixa à base da Barreira Frente. É também necessário alterar a altura total deste conjunto pois, a sua altura é inferior à dos rodízios que se encontram originalmente no carro de transporte.

HMI

A conceção de uma interface gráfica secundária incorporada na bancada móvel é também um dos objetivos desta dissertação. Esta interface seria utilizada maioritariamente aquando do processo de pingagem, apresentando informação relevante ao operador de forma cómoda.

Adicionalmente, foi pensada a existência de um sistema de comando manual do *track* para realizar a mudança de zona de projeção.

Assim, foi idealizada, inicialmente, a incorporação de um pequeno ecrã *touch*. Atendendo que a bancada acompanha o operador, este ecrã seria utilizado durante a totalidade do processo de pingagem, apresentando de forma cómoda todas as informações acerca dos componentes a pingar em cada zona de projeção. Adicionalmente, e aproveitando a funcionalidade *touch*, seriam implementados botões para realizar a mudança de zona de projeção do *track*.

Uma outra solução seria colocar a HMI da SARKKIS Robotics a bordo da bancada, centralizando, assim, todos os elementos de comando. Inicialmente, a HMI era um elemento fixo da célula, estando situada num local seguro e afastada dos elementos móveis.

As vantagens de colocar a HMI a bordo da bancada são:

- Centralização dos elementos de comando.
- Facilidade de acesso à totalidade da informação disponível em qualquer altura do processo produtivo.
- Não requer equipamento adicional, uma vez que a HMI já é um elemento existente na célula.

Contudo, esta segunda solução acresce na complexidade e quantidade de componentes móveis, que serão necessários ligar ao restante equipamento.

Em conjunto com a equipa da SARKKIS Robotics decidiu-se que a solução em que a HMI está a bordo da bancada seria a mais interessante, quer pelas vantagens que apresenta quer pelo nível de inovação que esta solução traz face ao que já foi realizado pela empresa em projetos anteriores.

Estrutura de suporte

Quanto à estrutura de suporte à HMI, esta pode ser rígida ou articulada.

Foi elaborada uma lista de soluções rígidas e articuladas, disponíveis no mercado ou feitas à medida, da qual, em conjunto com a equipa da SARKKIS Robotics, se selecionou a mais interessante. A compilação de imagens da Figura 46 mostra a solução escolhida de vários ângulos de visão.

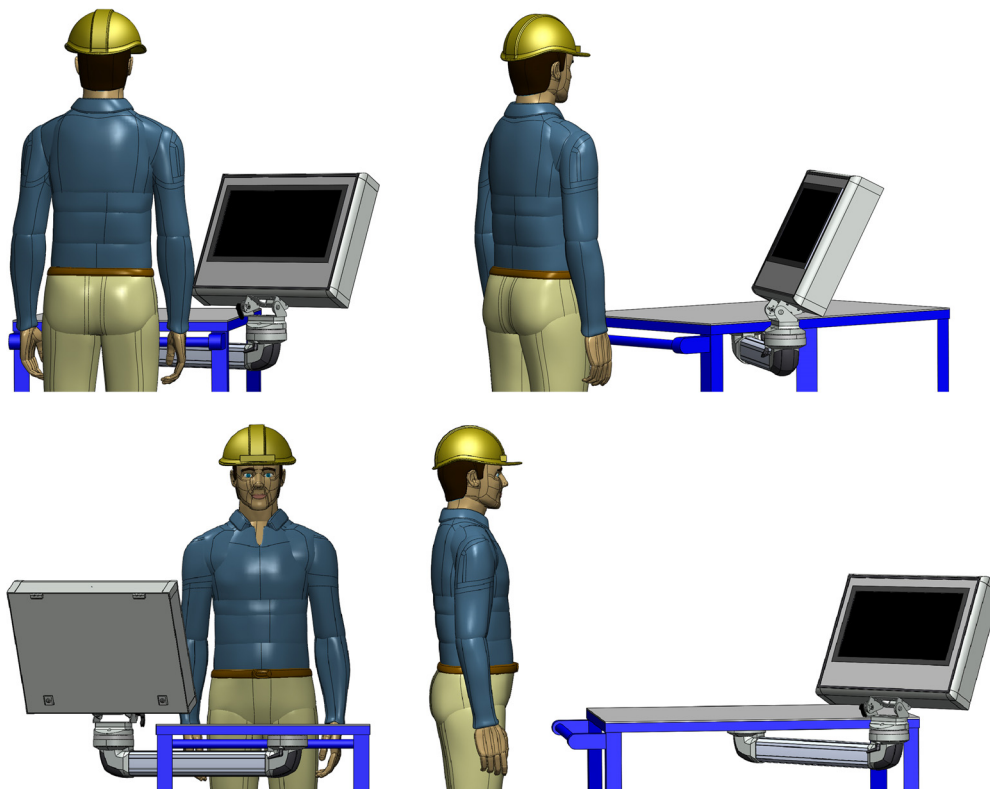


Figura 46 - Estrutura de suporte à HMI. Componentes Rittal [45].

A estrutura acima demonstrada é conseguida através de várias peças da marca Rittal [45]. A SARKKIS Robotics tem vindo a utilizar estruturas deste fornecedor em alguns dos seus projetos anteriores. Assim, foi dada alguma preferência a esta marca.

A solução ilustrada pelas imagens da Figura 46 representa uma configuração em que a estrutura articulada é apoiada sob o tampo, não ocupando nenhum espaço da superfície superior da bancada.

A HMI pode ser inclinada relativamente à vertical facilitando o contacto visual. Adicionalmente, pode ser rodada em torno de si própria e pode ser deslocada ao longo da bancada estando, contudo, limitada fisicamente pela estrutura da mesma.

No entanto, o facto de a HMI ficar deslocada da bancada pode causar dificuldades de passagem ao operador sempre que a bancada se encontre no interior da célula.

Em suma, esta é uma solução compacta e bastante flexível, permitindo que o operador a posicione rápida e facilmente para tornar a visualização cómoda em qualquer tarefa.

A Figura 47 e a Tabela 21 identificam e listam todos os componentes da estrutura. Os componentes nº Ref. 8 e 9 não são visíveis nesta figura.

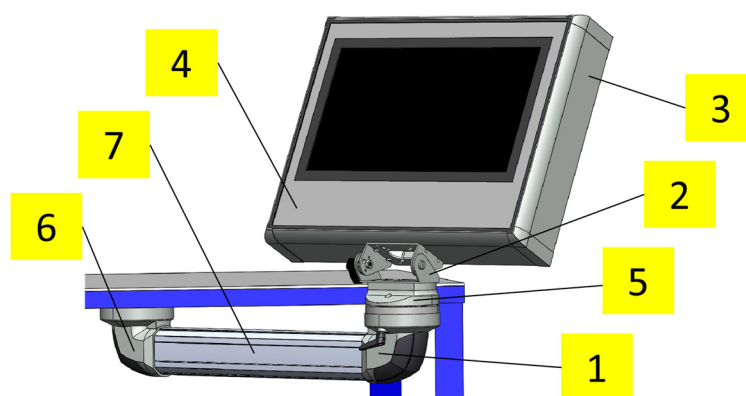


Figura 47 - Componentes da estrutura Rittal da HMI [45].

Tabela 21 – Configuração nº1: Componentes da estrutura Rittal da HMI

Nº Ref	Código	Nome do componente	Quantidade
1	6206.380	Acoplamento CP60 90º articulado	1
2	6206.460	Adaptador de inclinação	1
3	6380.400	Optipanel 520 × 400 × 100	1
4	6028.014	Placa frontal com pinos roscados M5	1
5	6212.500	Adaptador CP120/60	1
6	6206.700	Suporte CP60 articulado	1
7	6206.050	Tubo CP60 500mm	1
8	6206700	Porcas elásticas M5	1
9	6053.500	Kit de fixação para Optipanel	1

4.1.6 Medidas de proteção adicionais

As normas ISO 10218-1 [17] e ISO 10218-2 [7] especificam algumas regras e medidas de proteção relativas a robôs e a células robóticas.

Paragem de emergência

Cada estação de comando capaz de iniciar movimento ou outra função perigosa deve possuir um meio manual de paragem de emergência. A paragem de emergência deve parar o movimento de todos os componentes perigosos. Estes dispositivos devem estar bem visíveis.

Assim, será colocada uma botoneira de emergência na proximidade da HMI.

Modos de funcionamento

Todos os elementos de comando que precisarem de ser utilizados durante operações automáticas devem estar localizados no exterior do espaço protegido, forçando a pessoa que os está a utilizar a estar fora da zona protegida.

Assim, será instalado um seletor manual de modo de funcionamento no exterior da célula.

Para além disso, todos esses controlos devem estar localizados numa posição em que tanto a máquina como o espaço protegido sejam visíveis.

Uma vez em modo automático, a entrada no espaço protegido deve levar à paragem de todos os equipamentos capazes de gerar perigo.

Em modo manual a velocidade reduzida, a velocidade do TCP selecionado não deve ultrapassar 250mm/s. Tal como as consolas do robô, qualquer dispositivo capaz de controlar o robô deve:

- Operar em modo manual com velocidades reduzidas.
- Estar munido de um *enabling device* de três posições.
- Possuir um elemento com função de paragem de emergência.
- Não pode ser capaz de iniciar uma operação automática.

Contudo, existem exceções a estas condições descritas nas normas acima indicadas.

Uma vez que se pretende operar o *track* e movimentar os Viradores, quando operador se encontra no interior da célula, a HMI deve respeitar as condições acima indicadas. Ou seja, deverá ser integrado um *enable device* e uma botoneira de emergência na proximidade da HMI. A Figura 48 mostra estes dispositivos integrados na caixa da HMI.

A Tabela 22 sumariza todos os dispositivos de segurança adicionais necessários.

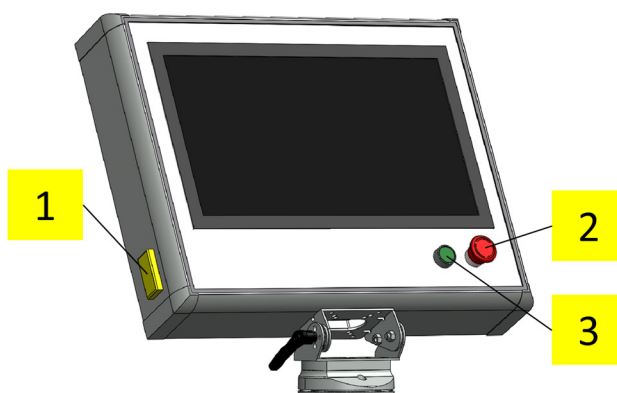


Figura 48 - Dispositivos de proteção da HMI: 1-Botão *Enable*; 2-Botoneira de emergência; 3-Botão de rearme.

Tabela 22 – Configuração nº1: dispositivos de segurança adicionais

Dispositivo	Quantidade	Descrição
Botões de emergência	2	Colocados um à entrada da célula e outro na caixa da HMI
Seletor de duas posições	1	Colocado à entrada da célula
<i>Enable device</i> de 3 posições	1	Colocado na caixa da HMI

4.1.7 Procedimento operacional - versão final

Neste subcapítulo são detalhadas todas as etapas que constituem o procedimento de operação da máquina, versão final. As diferenças deste procedimento, comparativamente com o procedimento inicial, descrito no subcapítulo 4.1.2, estão sublinhadas.

Carregamento

A célula deve estar totalmente parada e selecionado o modo manual.

A viga é transportada por uma ponte rolante através da barreira luminosa.

O robô e os Viradores têm de estar na posição *home*, de modo a que a viga possa ser descarregada da ponte rolante.

O operador acompanha a operação de transporte e posicionamento da viga nos cavaletes, ficando a viga disposta paralelamente ao *track* do robô.

O abastecimento do carrinho auxiliar com todas as peças necessárias deve ser feito neste momento.

Calibração

O operador deve sair da zona protegida, selecionar o modo automático e iniciar, através da HMI, a calibração do sistema.

Pingagem

Após a calibração do sistema, têm início os processos de projeção e de pingagem de componentes à viga.

O projetor é posicionado na 1ª zona de projeção, onde é feita a projeção. A célula deve, agora, entrar em modo manual, permitindo que o operador e a bancada móvel entrem na zona protegida, através da barreira luminosa.

É feito, pelo operador, o posicionamento e consecutiva pingagem das peças na posição correta, segundo a projeção.

Após pingagem de todos os componentes desta zona de projeção, o operador deve recolher o seu equipamento (equipamento de soldadura) e dirigir-se à bancada.

Mudança de zona de projeção

Na bancada existem botões que habilitam o deslocamento do conjunto projetor-robô entre as várias zonas ou posições de projeção, permitindo continuar a projetar e pingar peças ao longo da viga, garantindo, contudo, que o operador se encontra em segurança a pressionar o botão *enable device* e não exposto ao perigo.

O operador deve manter contacto visual com a máquina durante esta etapa do processo.

Pingagem

Estando todas as peças a soldar pingadas na sua posição, a etapa de preparação está terminada. O operador sai do espaço protegido, atravessando a barreira luminosa, trazendo com ele a bancada móvel.

Soldadura

Estando fora do espaço protegido o operador deve comutar, novamente, para o modo automático. No HMI, o operador dá início à soldadura automática de todos os componentes.

No final da operação, o robô deve ficar “arrumado” na sua posição *home*.

Nota:

Existem duas possibilidades:

- pinga-se e solda-se duas faces, roda-se a viga, pinga-se e solda-se as duas faces restantes. Nesta possibilidade, a viga é rodada uma única vez;
- pinga-se as quatro faces e só depois se solda. Neste caso, fazem-se duas rotações à viga, uma durante a pingagem e outra durante a soldadura.

Controlo de qualidade

Finalizando o processo de soldadura, o operador deve seleccionar o modo manual e proceder ao controlo de qualidade.

Descarregamento

Durante esta etapa a célula deve estar totalmente parada.

O operador fixa a viga à ponte rolante e a viga é transportada para o exterior da célula.

4.2 PLC de segurança

Neste subcapítulo, é descrito todo o processo de controlo dos equipamentos de segurança e de como a célula e os respetivos modos de funcionamento (manual e automático) são implementados.

4.2.1 Escolha do equipamento

De seguida, é feito um sumário de todas as entradas e saídas que o PLC de segurança a ser escolhido deve apresentar para monitorizar e controlar todos os equipamentos de segurança.

Entradas

- 2 botões de emergência – 4 entradas
- 2 botões de rearme – 2 entradas
- Barreira ótica – 2 entradas
- Botão *enable* – 4 entradas
- Seletor de modo – 2 entradas

Total = 14 entradas

Saídas

- Emergência Robô – 2 saídas OSSD
- Emergência Viradores – 2 saídas OSSD
- Emergência *soft*PLC – 1 saída
- Botão *enable* (sinalização) – 1 saída

Total = 6 saídas (das quais 4 são OSSD)

Em projetos anteriores, a SARKKIS Robotics recorreu a PLC's de segurança da marca ReeR [42], tendo obtido bons resultados. Assim, foi dada preferência a esta marca e foram escolhidos os equipamentos a seguir apresentados.

- ReeR Mosaic M1 – PLC de segurança
 - 8 entradas digitais
 - 2 OSSD
 - 2 saídas programáveis
- ReeR MI8O2 – Unidade de expansão
 - 8 entradas digitais
 - 2 OSSD
 - 2 saídas programáveis
- ReeR MR2 – Relé de segurança
 - 2 contactos normalmente abertos
 - 1 contacto normalmente fechado

O relé de segurança MR2 fornece as saídas a relé necessárias para a interação com a carta de segurança do controlador do robô.

4.2.2 Programação e funcionalidades do programa

Os PLC da ReeR são programados utilizando uma linguagem proprietária semelhante aos diagramas de blocos. A Figura 49 ilustra o programa construído. Para facilitar a compreensão e a descrição do programa, este foi dividido em cinco partes.

- Parte 1 – Seletor de modo e botão *enable*
- Parte 2 – Botoneira de emergência exterior e barreira luminosa
- Parte 3 – Botoneira de emergência da HMI
- Parte 4 – Consola em modo *teach*
- Parte 5 – Saídas e sinalização

A célula pode operar segundo dois modos de funcionamento: modo manual e modo automático. O modo de funcionamento é selecionado através do seletor de modo representado pelo *switch* Manual/Auto na Parte 1.

Na Parte 5 estão as saídas que habilitam ou desabilitam o funcionamento da célula. Assim, quando estas estão acionadas (a verde, como na Figura 49), todos os equipamentos podem ser utilizados; quando estão desabilitadas (a vermelho) a célula entra em emergência.

Uma vez em modo automático, como na Figura 49, existem três equipamentos capazes de despoletar emergência: a botoneira de emergência, que se encontra fixa no exterior da célula (E-STOP ChA e ChB da Parte 2), a barreira luminosa, quando cruzada (ESPE da Parte 2), e a botoneira de emergência da HMI (E-STOP ChA e ChB da Parte 3).

Ambas as Partes 2 e 3 contemplam equipamentos de segurança para o funcionamento em modo automático e, em cada uma delas, existe uma botoneira de Rearme (SWITCH Rearme). Uma vez em emergência, o Rearme da Parte 2 apenas habilita as saídas se estas tiverem sido desabilitadas por um elemento de proteção da Parte 2; o mesmo acontece para o Rearme da Parte 3 que apenas habilita as saídas se estas tiverem sido desabilitadas pela botoneira de emergência da Parte 3.

A funcionalidade descrita pelo parágrafo anterior foi implementada para não ser possível o rearme a partir do interior da célula, ou seja, considerando a célula em modo automático e o operador fora do espaço protegido, se este tentar entrar com o carrinho na célula, cruzando a barreira luminosa, a célula, tal como esperado, entrará em emergência. Neste momento, o operador encontra-se com a bancada móvel no interior da célula e a única maneira de sair de emergência é através da botoneira de Rearme, que se encontra no exterior da célula (Rearme da Parte 2), pois a botoneira de rearme que se encontra na HMI (junto ao operador no interior da célula e representado pelo SWITCH da Parte 3) não permite rearmar a célula quando o estado de emergência foi despoletado por um dos elementos da Parte 2.

Em modo manual, a célula, para todos os efeitos, encontra-se em emergência. Neste modo, as saídas só são habilitadas quando o botão *enable* é pressionado na posição central (ENABLE ChA e ChB da Parte 1).

Adicionalmente, em modo manual, é possível habilitar apenas a movimentação do robô, recorrendo a um sinal proveniente da consola do robô (representado pelo SWITCH da Parte 4), significando que esta se encontra em modo *teach*.

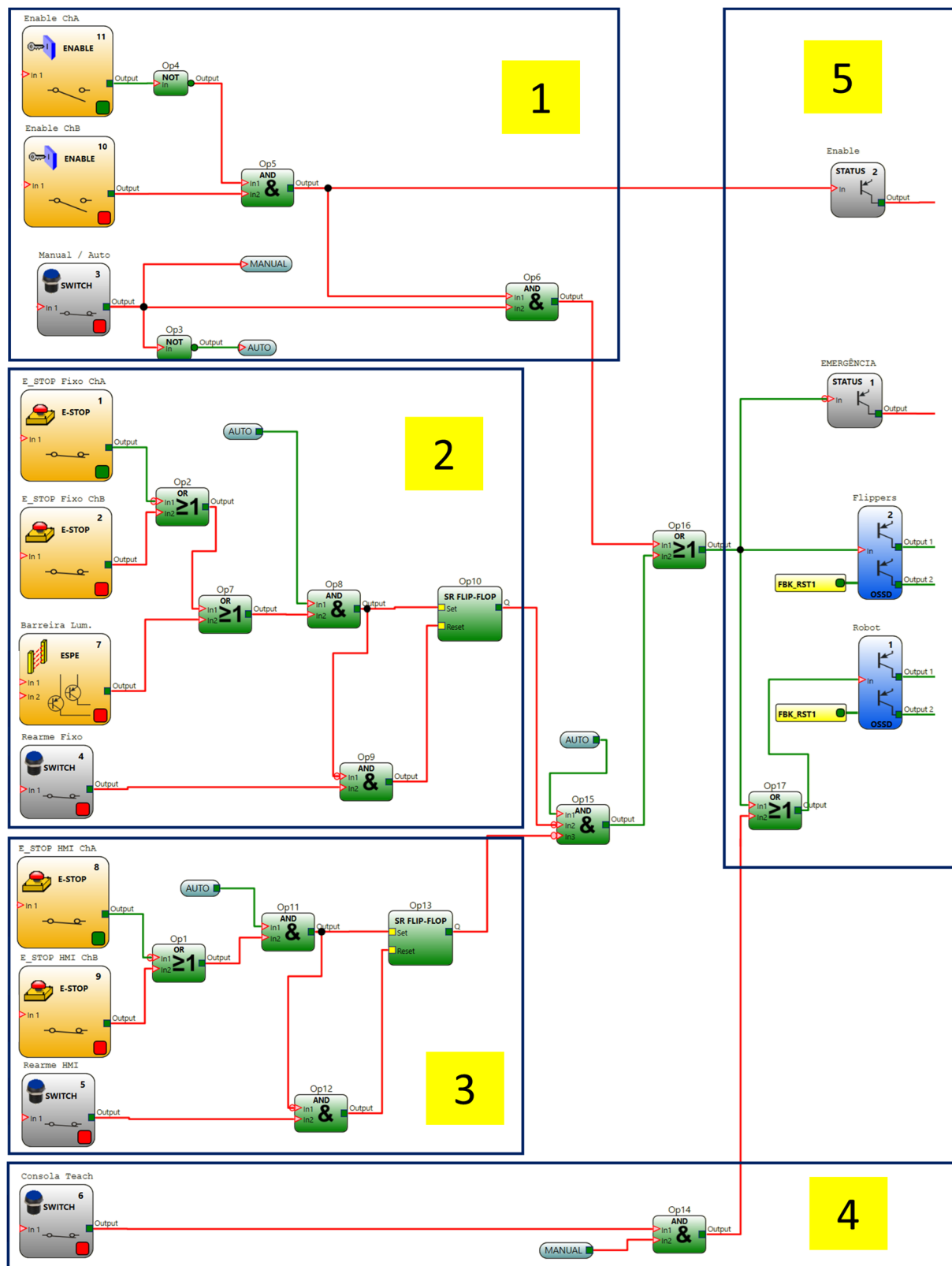


Figura 49 - Programa do PLC de Segurança.

4.3 Tarefa colaborativa - *Hand guiding*

Este capítulo retrata um problema que será recorrente durante o processo de pingagem de componentes à viga. De modo a compreender a sua origem, é apresentado um exemplo demonstrativo.

Seguidamente, é proposta uma possível solução que recorre à interação do operador diretamente com o robô, constituindo, assim, uma tarefa colaborativa. De acordo com as normas disponíveis, é elaborado um conjunto de requisitos para que esta tarefa se realize em segurança.

4.3.1 Exemplo demonstrativo do problema

Considera-se a parte da célula ilustrada na Figura 50.

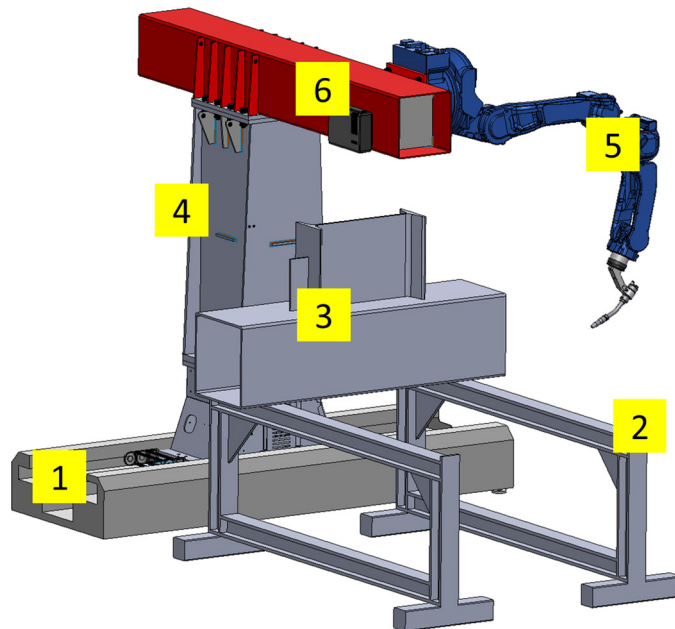


Figura 50 - Célula robótica parcial: 1-Track; 2-Cavalete auxiliar; 3-Componentes a soldar; 4-Semipórtico; 5-Robô; 6-Projetor laser.

Pretende-se simular o processo de soldadura automática de componentes numa viga. Seguem-se as várias etapas deste processo de modo a exemplificar o problema.

A Figura 51 mostra os componentes a soldar. Pretende-se soldar o perfil nº2 na viga nº1 e, de seguida, soldar a chapa nº3 a estes dois componentes.

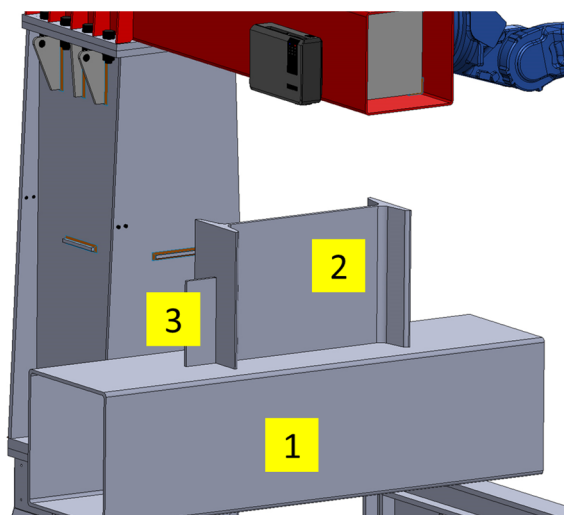


Figura 51 - Célula robótica parcial: Componentes a soldar.

O processo inicia-se pela projeção, na viga nº1, do local onde o operador deve pingar os componentes nºs 1 e 2. A Figura 52 mostra a projeção efetuada.

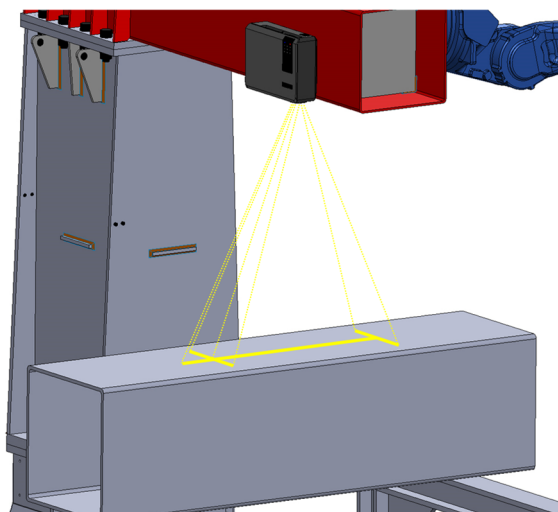


Figura 52 - Célula robótica parcial: Projeção do perfil nº2 e da chapa nº3.

Após o operador ter pingado o perfil nº2 no local devido, segue-se a pingagem da chapa nº3.

Contudo, nesta etapa, o operador é incapaz de realizar o pingamento, pois a projeção do local de pingagem da chapa nº3 ficou bloqueada pelo perfil nº2. A Figura 53 ilustra esta situação.

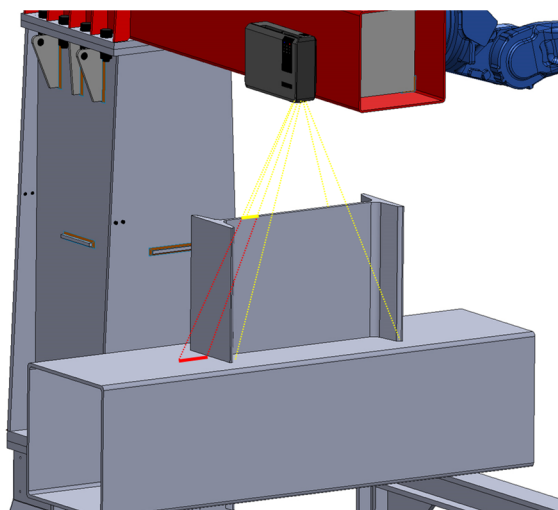


Figura 53 - Célula robótica parcial: Projeção falhada da chapa nº3.

O problema que ocorreu no exemplo acima demonstrado pode, em certas circunstâncias, ser evitado, pois o operador podia ter pingado primeiro a chapa nº3 e só depois avançar para o perfil nº2. Contudo, o exemplo acima é de carácter meramente demonstrativo e apenas pretende demonstrar o problema.

Na realidade, mesmo que a perspicácia do operador seja apurada, é certo que situações semelhantes a esta irão ocorrer.

De seguida, é apresentada uma solução conceptual para este problema.

4.3.2 Solução colaborativa

A forma mais expedita de resolver este problema é o ajuste da zona de projecção de forma a colocar o projetor numa posição mais favorável.

Para iniciar o processo de mudança de projecção, o operador tem de se deslocar da zona de pinagem, a verde na Figura 31, até à bancada móvel onde se encontra a HMI.

A introdução de uma tarefa colaborativa evita esta deslocação, permitindo que o operador, a partir do seu local de pingagem, possa interagir directamente com o robô (movimentação do *track*), dentro da sua área de trabalho, a vermelho na Figura 31. Esta tarefa colaborativa insere-se na categoria de *Hand Guiding* e, como tal, deve cumprir os seguintes requisitos:

- A operação colaborativa deve estar visualmente sinalizada
- A interface de controlo deve estar situada perto do robô (*end-effector*) e estar equipada com os seguintes elementos:
 - Botoneira de emergência
 - Botão *enabling* de três posições

5 Projeto de automação dos Viradores

O presente capítulo será dedicado à apresentação dos Viradores na sua vertente estrutural e hidráulica e do seu princípio de funcionamento. O conhecimento destes aspetos é fundamental para o sucesso na programação do equipamento.

Será também apresentada e descrita a estratégia de programação utilizada e a solução de sensorização adotada.

5.1 Virador - Machinery Services

Os Viradores utilizados são semelhantes ao Ocean Flipper apresentado no estado da arte desta dissertação, subcapítulo 2.3.3. É um equipamento rápido e extremamente compacto, o que é uma mais valia para o projeto, pois como o robô está suspenso num semipórtico móvel, em operações automáticas os Viradores não constituem um obstáculo ao seu movimento.

Este equipamento é constituído por duas pás que podem rodar segundo um eixo de rotação comum. Cada pá tem acoplado um atuador hidráulico responsável pelo seu movimento de rotação. Todos os movimentos do Virador são conseguidos através de atuadores hidráulicos acionados por electroválvulas direcionais.

O conjunto das duas pás e respetivo sistema de acionamento estão suportados por um chassi móvel, permitindo deslocar o conjunto das duas pás paralelamente ao chão e perpendicularmente ao *track* (eixo linear do robô). Este movimento é efetuado com recurso a um terceiro atuador hidráulico de maior curso.

A imagem seguinte, Figura 54, é uma fotografia dos Viradores durante a montagem da célula.

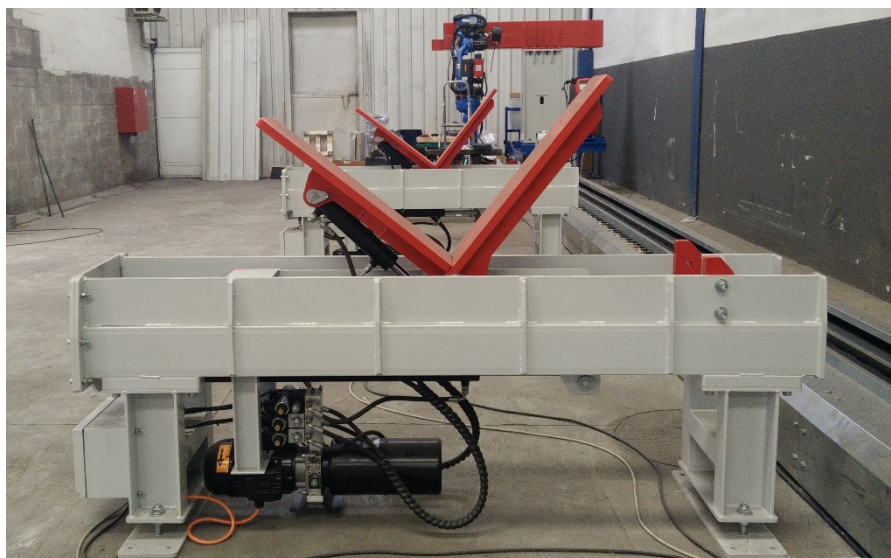


Figura 54 - Fotografia dos Viradores durante a fase de montagem da célula.

Nos próximos subcapítulos é feita uma análise mais detalhada dos componentes e do funcionamento dos Viradores. Assim o subcapítulo 5.1.1 tratará da componente estrutural, o subcapítulo 5.1.2 tratará, apenas, da componente hidráulica e o subcapítulo 5.1.3 descreverá o princípio de funcionamento do Virador.

5.1.1 Componente estrutural

Neste subcapítulo será tratada e analisada apenas a componente estrutural do Virador, a componente hidráulica será analisada no subcapítulo seguinte, subcapítulo 5.1.2.

As imagens seguintes, Figura 55 e Figura 56, são representações CAD do Virador. Na Figura 56 foram escondidas algumas partes do Virador para facilitar a compreensão do seu funcionamento. Além disso, a central hidráulica montada sob o Virador, visível na Figura 54, não aparece nestas representações CAD.

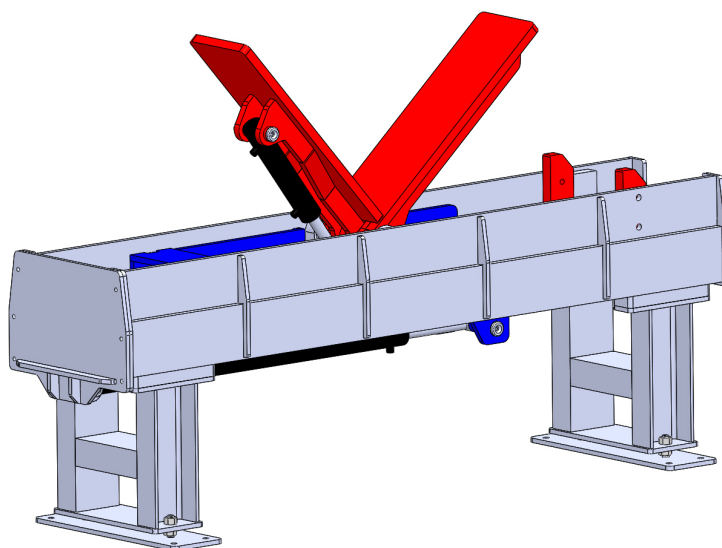


Figura 55 - Representação CAD do modelo do Virador.

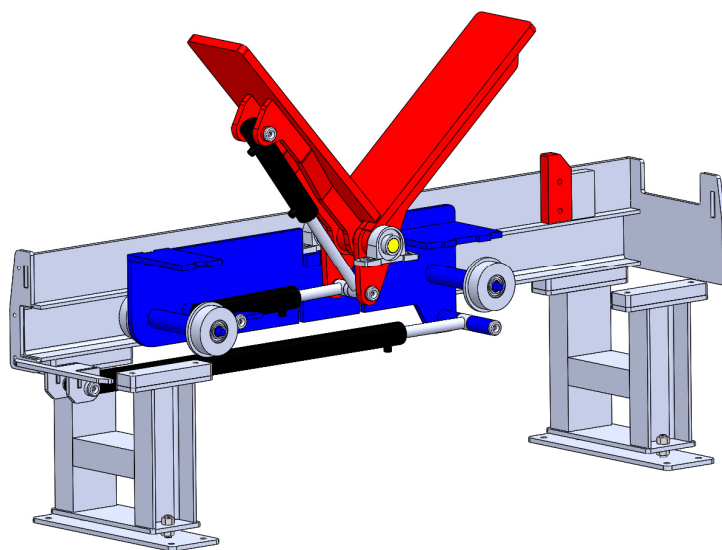


Figura 56 - Representação CAD detalhada do modelo do Virador.

Consegue-se dividir toda a estrutura em duas partes: a Carruagem interior e a Carcaça exterior, que são analisadas de seguida.

Carruagem interior

A figura abaixo, Figura 57, é uma representação CAD da Carruagem interior, em que, mais uma vez, algumas partes foram escondidas para uma melhor visualização de todos os componentes.

A Carruagem interior é composta pelos seguintes componentes principais:

- Chassi metálico – a azul
- 2 Pás – Pá esquerda e Pá direita
- Atuador A e B – cada um associado a uma das pás
- Atuador C – associado ao movimento da Carruagem
- Veio – a amarelo e coincidente com o eixo de rotação das pás
- 4 rodas metálicas

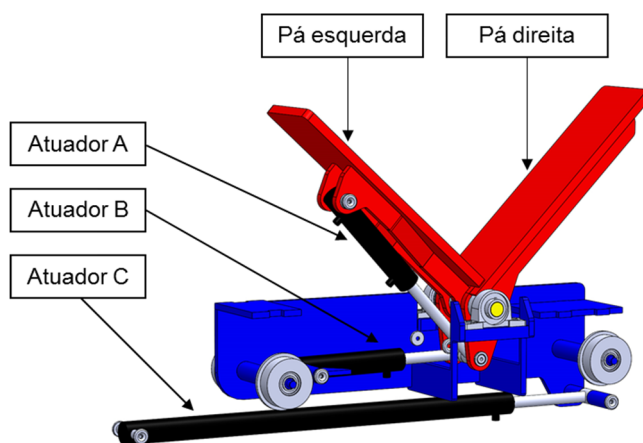


Figura 57 - Representação CAD da Carruagem interior.

Como se pode observar na Figura 57, ambas as pás têm o seu eixo de rotação coincidente. Este eixo de rotação é também coincidente com o veio que se encontra a amarelo. Por sua vez, este veio é fixo ao chassi, por meio de chumaceiras.

O Atuador A tem uma das extremidades fixa na pá esquerda e a outra fixa na estrutura da Pá direita, ou seja, o avanço e o recuo deste atuador faz variar o ângulo da Pá esquerda relativamente à Pá direita. O valor deste ângulo pode variar entre 0° e 90°.

Quanto ao Atuador B, uma extremidade é fixa ao chassi e outra é fixa na estrutura da Pá direita. Assim, este atuador controla o ângulo da Pá direita relativamente ao chassi da carruagem. É de notar que o valor do ângulo das duas pás mantém-se fixo, pois é controlado pelo Atuador A, ou seja, avançando apenas o Atuador B estaremos a aumentar o ângulo da Pá direita e a diminuir, na mesma proporção, o ângulo da Pá esquerda, relativamente ao chassi.

As 4 rodas desta carruagem são o elemento de contacto com a carcaça exterior e têm a função de guiar e facilitar o movimento horizontal da carruagem no interior da Carcaça exterior. O movimento é controlado pelo Atuador C, que tem uma extremidade fixa ao chassi e a outra fixa na Carcaça exterior (perceptível na Figura 56).

Carcaça exterior

A Figura 58 é uma representação CAD da carcaça exterior, na qual foram colocadas algumas superfícies transparentes para se poder ver os detalhes interiores da carcaça.

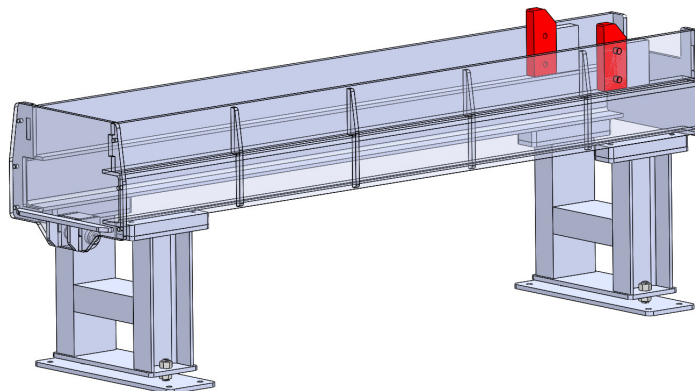


Figura 58 - Representação CAD dos detalhes interiores da carcaça exterior.

Das figuras anteriores, depreendem-se como principais elementos funcionais da carcaça:

- 2 batentes – a vermelho.
- 2 carris – para guiamento da carruagem
- Apoio do Atuador C
- 2 pés reguláveis

De modo a minimizar os erros e as variáveis durante o processo de soldadura, a viga deve ser colocada, tão bem quanto possível, paralela ao *track* e ao solo.

O sistema de nivelamento consiste num sistema de porca e contra porca num perno roscado, sendo que em cada pé existem dois destes sistemas, ou seja, cada Virador tem quatro pontos de ajuste em altura, garantindo, assim, o seu nivelamento relativamente ao solo. Estes sistemas de nivelamento estão representados nos desenhos CAD das Figura 55, Figura 56 e Figura 58

Os dois batentes presentes em cada Virador visam garantir o paralelismo com o *track*.

5.1.2 Componente hidráulica

Todo o equipamento hidráulico, com exceção dos atuadores e respetivas manguueiras, está montado por baixo da carcaça, como é possível ver na Figura 54.

De seguida será apresentado:

- Esquema hidráulico do Virador – Figura 59;
- Listagem todos os componentes hidráulicos do Virador com descrição de alguns detalhes funcionais – Tabela 23;
- Fotografias legendadas da central hidráulica do Virador – Figura 60.

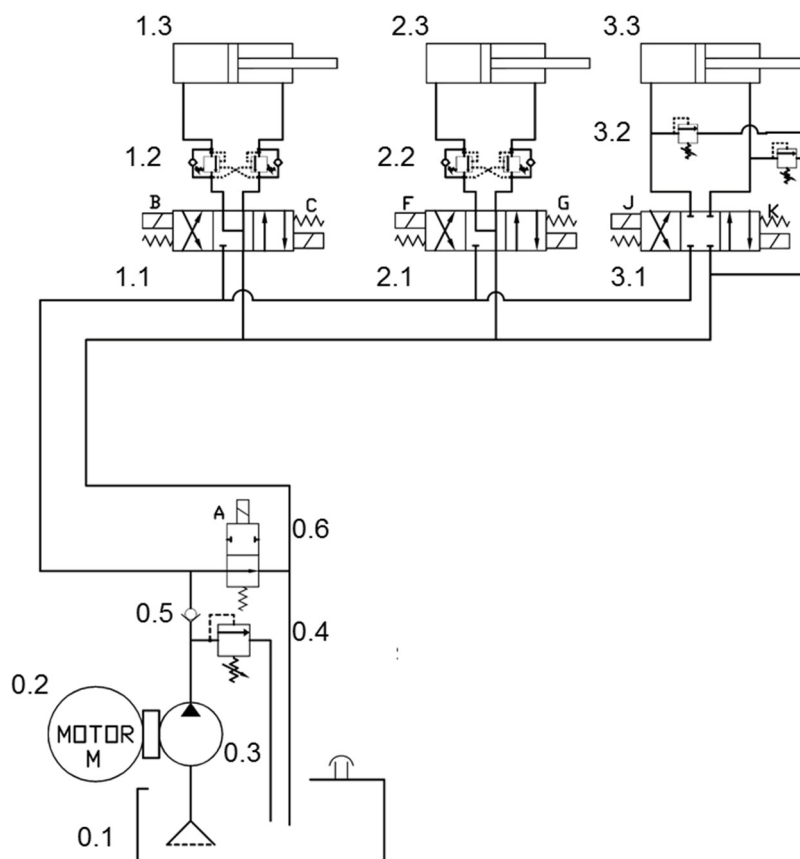


Figura 59 - Esquema hidráulico simplificado do Virador.

Tabela 23 – Lista dos componentes hidráulicos do Virador

Ref.	Denominação	Descrição
0.1	Reservatório	
0.2	Motor trifásico	
0.3	Bomba hidráulica	
0.4	Válvula limitadora de pressão	É regulável e limita a pressão máxima no circuito
0.5	Válvula de retenção simples assento axial	
0.6	Eletroválvula de descarga	Retorno por mola
1.1	Eletroválvula direcional 4/3 centragem por mola	Digital, centro em Y com A e B ligado a T
1.2	Grupo de 2 válvulas de sequência de pilotagem externa cruzada com função de contrapressão	Em operação têm função de válvulas de sequência; com o sistema parado têm função de contrapressão
1.3	Atuador hidráulico de duplo efeito assimétrico	Responsável pela rotação da pá esquerda

2.1	Eletroválvula direcional 4/3 com centragem por mola	Digital. Centro em Y com A e B ligado a T
2.2	Grupo de 2 válvulas de sequência de pilotagem externa cruzada com função de contrapressão	Em operação têm função de válvulas de sequência; com o sistema parado têm função de contrapressão
2.3	Atuador hidráulico de duplo efeito assimétrico	Responsável pela rotação da pá direita
3.1	Eletroválvula direcional 4/3 com centragem por mola	Digital, centro fechado
3.2	Grupo de 2 válvulas limitadoras de pressão	Limitam a pressão nas camaras do atuador 3.3
3.3	Atuador hidráulico de duplo efeito assimétrico	Responsável pelo movimento horizontal da carruagem

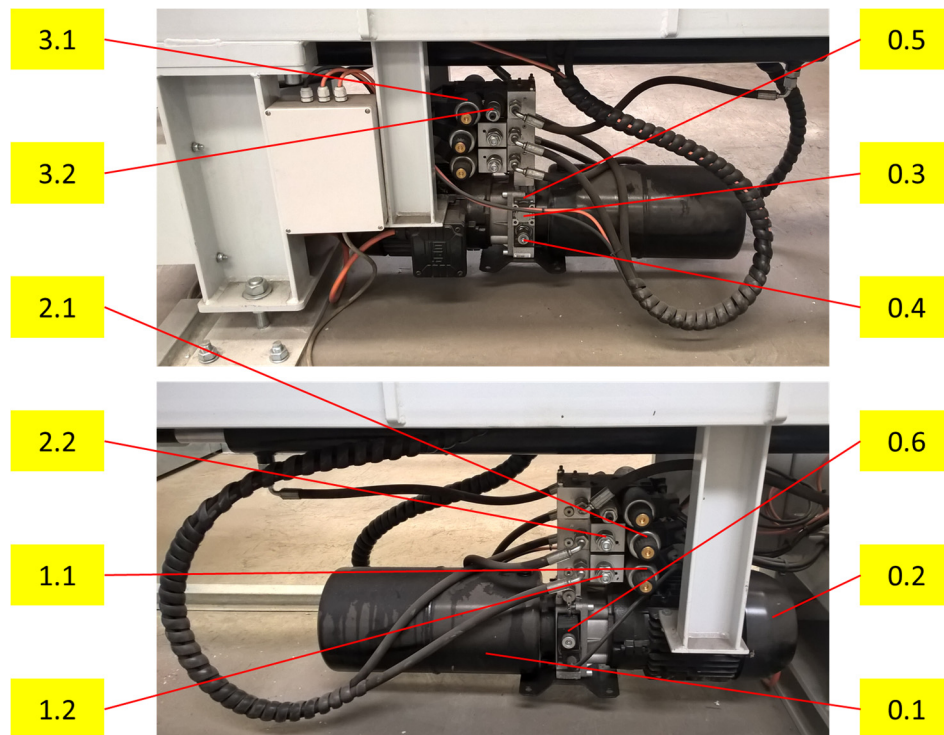


Figura 60 – Fotografia da central hidráulica do Virador.

5.1.3 Princípio de funcionamento

Neste subcapítulo serão descritos todos os passos necessários para a rotação e para o movimento horizontal da viga.

Rotação

A sequência de imagens que se segue, Figura 61 à Figura 64, representa as várias etapas de uma rotação de 90° no sentido positivo (considera-se sentido positivo o sentido contrário aos ponteiros do relógio).

Etapa 0 – Etapa inicial. Ambas as pás estão a 0° com o chassi. A viga está colocada à direita do eixo de rotação das pás.

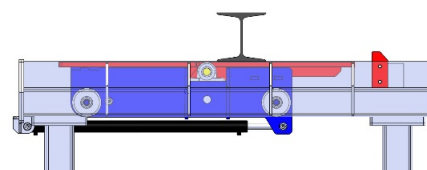


Figura 61 – Rotação Positiva Etapa 0.

Etapa 1 – Avançando o Atuador A, a Pá esquerda sobe até ficar perpendicular com a Pá direita.

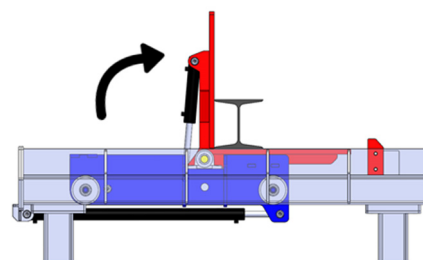


Figura 62 - Rotação Positiva Etapa 1.

Etapa 2 – Avançando apenas o Atuador B, a Pá direita fica a 90° e, consecutivamente a Pá esquerda fica a 0° com o chassi. A viga está agora rodada 90°.

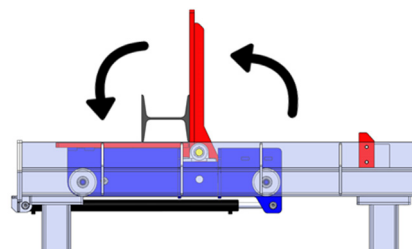


Figura 63 - Rotação Positiva Etapa 2.

Etapa 3 – Recuando agora ambos os Atuadores A e B, sem movimentar a viga, as pás tomam a sua posição inicial.

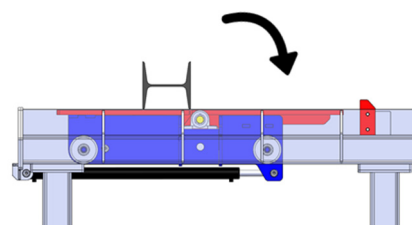


Figura 64 - Rotação Positiva Etapa 3.

A rotação no sentido negativo (sentido dos ponteiros do relógio) dá-se de forma semelhante à rotação no sentido positivo, contudo, é de destacar algumas diferenças nas etapas iniciais.

Etapa 0 – Etapa inicial. Neste caso, difere da etapa inicial da rotação positiva por a viga se encontrar à esquerda do eixo de rotação das pás.

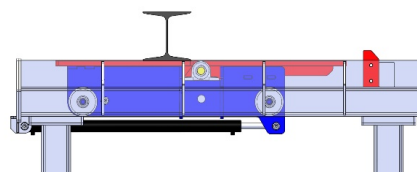


Figura 65 - Rotação Negativa Etapa 0.

Etapa 1 – Avançando os Atuadores A e B em simultâneo, a Pá direita sobe até chegar à posição vertical.

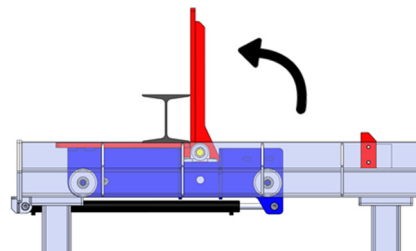


Figura 66 - Rotação Negativa Etapa 1.

As etapas seguintes, Etapa 2 e 3, seguem o mesmo raciocínio que na rotação no sentido positivo.

Movimento horizontal

Quando ambas as pás estão a 0° , a viga encontra-se apoiada sobre a carcaça exterior, pois as faces das pás, nesta posição, ficam a uma cota inferior à das paredes laterais da carcaça exterior.

Desta forma, para ser possível movimentar a viga sem haver arrastamento é necessário realizar a suspensão da mesma. A suspensão da viga é feita utilizando ambas as pás, sendo que esta deixa de estar em contacto com a carcaça desde o momento em que a Pá direita faz um ângulo 30° com o chassi.

As figuras que se seguem, Figura 67 e Figura 68, descrevem as etapas necessárias para a realização do movimento.

As etapas 0 e 1 deste movimento são as mesmas que as etapas 0 e 1 para a rotação da viga, Figura 61 e Figura 62. Ficando o equipamento numa posição semelhante à da Figura 9.

Etapa 2 – Avançando apenas o Atuador B, a viga fica suspensa quando o ângulo entre a Pá direita e o chassi está compreendido entre 30° e 150° .

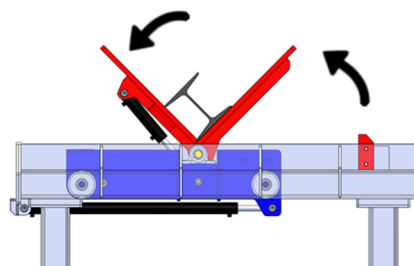


Figura 67 - Mov. Horizontal Etapa 2.

Etapa 3 – Acionando o Atuador C, é possível movimentar a viga em qualquer um dos sentidos.

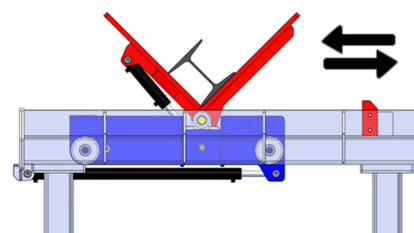


Figura 68 - Mov. Horizontal Etapa 3.

As etapas 4 e 5 deste movimento são o inverso das duas primeiras etapas (etapas 0 e 1), ficando o equipamento numa posição semelhante à posição inicial, Figura 61.

5.1.4 Limitações

As dimensões máximas e mínimas admitidas para o comprimento da viga são, respetivamente, 12m e 4m. O comprimento máximo está limitado pelo curso do *track*, que permite a movimentação do robô ao longo da viga, e o comprimento mínimo está limitado pela distância entre os Viradores, uma vez que para movimentar ou rodar uma viga são sempre necessários os dois Viradores.

A largura da secção da viga também é limitada e deve ter uma dimensão compreendida entre 80 e 600mm. O curso do Atuador C é de 810mm. A diferença entre este a o valor máximo da largura da viga corresponde à margem que o Virador necessita para conseguir mover e rodar a viga. O valor mínimo para a largura da menor viga é limitado pela menor distância entre a Pá esquerda, na posição vertical, e os batentes. Assim, só se garante que a viga é corretamente encostada aos batentes se esta apresentar uma largura mínima igual ou superior a 80mm.

5.2 Controlo dos Viradores

Uma vez que na célula já existe um PC, utilizado pelo *software* de controlo do robô, este foi utilizado, também, como *softPLC*, usufruindo, assim, de todas as vantagens deste tipo de equipamento, como foi descrito no estado da arte desta dissertação.

Para a implementação desta solução será usado o CodeSys da 3S-Smart Software Solutions GmbH [46] como ambiente de desenvolvimento para programação do *softPLC* (3S-Smart) em conjunto com um módulo de IOs da WAGO.

A parte de programação e todas as funcionalidades programadas são abordadas no capítulo 5.4.

5.2.1 Módulo de IO's

O módulo de IO's disponibilizado é o WAGO 750-352, sendo este apenas um *fieldbus coupler* ao qual se podem adicionar diferentes módulos.

Segue-se uma lista que descreve os módulos necessários, Tabela 24:

Tabela 24 - Lista de módulos WAGO necessários

Módulo	Qtd.	Descrição	Posição Bus
750-602 – <i>Power Supply</i>	2	Alimenta os módulos à sua direita e isola a alimentação dos módulos que lhe são adjacentes Este isolamento permite separar e proteger os equipamentos mais sensíveis e dispendiosos	-
750-402 – 4 <i>Digital Input</i>	2	Monitorização do estado do PLC de segurança e o funcionamento das centrais hidráulicas dos Viradores	1 e 2
750-453 – 4 <i>Analog Input</i>	2	Ver subcapítulo 5.3.3	3 e 4
750-504 – 4 <i>Digital Output</i>	1	Acionamento das centrais hidráulicas, sinalização do estado do PLC de segurança e baliza de sinalização de modo	5
750-508 – <i>Digital Output</i>	8	Alimentação dos solenoides dos Viradores	6 a 13
750-600 – <i>End Module</i>	1	Módulo terminal	14

A Figura 69 ilustra os módulos utilizados. Os *Power Supply* não são visíveis pois não fazem parte do *Bus* de comunicação do equipamento.

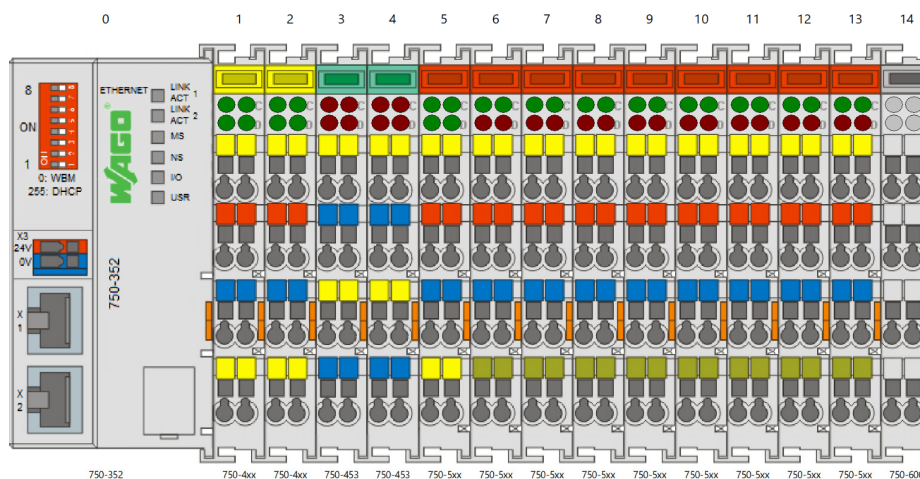


Figura 69 - Módulos WAGO.

Configuração do Módulo de IO's

O módulo 750-352 é um *fieldbus coupler* que suporta os protocolos de comunicação Ethernet/IP, ou *Ethernet Industrial Protocol*, e Modbus TCP/UDP.

Estes *couplers* já foram utilizados pela SARKKIS Robotics em projetos anteriores, usando sempre o protocolo Modbus para comunicação com o seu *software*. Uma vez que o CodeSys suporta os dois protocolos, optou-se por utilizar o protocolo Ethernet/IP, estudando e documentando a sua integração para utilização em futuros projetos.

Configuração nos módulos Wago

Depois de montados todos os módulos necessários, estes devem ser configurados para funcionamento sobre Ethernet/IP da seguinte forma:

- Através dos *DIP switches* presentes no coupler escolhe-se o endereço de IP para os módulos.
- Abrir a página web do dispositivo digitando o IP anteriormente selecionado na barra de endereços do *browser*.
- Desativar os protocolos de ModBus que vêm selecionados por defeito (ModBus UDP e TCP) e ativar o protocolo Ethernet/IP.

Para efetivar as alterações os módulos devem ser reiniciados.

Configuração no Codesys

No CodeSys, para configurar corretamente os módulos WAGO na utilização de Ethernet/IP é necessário:

- Adicionar o adaptador de *Ethernet* e abrir a respetiva Tab, escolher a interface que será utilizada (placa de rede do PC, placa de Wi-Fi ou outra). De seguida, introduz-se o IP (fixo) do PC onde correrá o *softPLC*. A Figura 70 ilustra este primeiro passo.

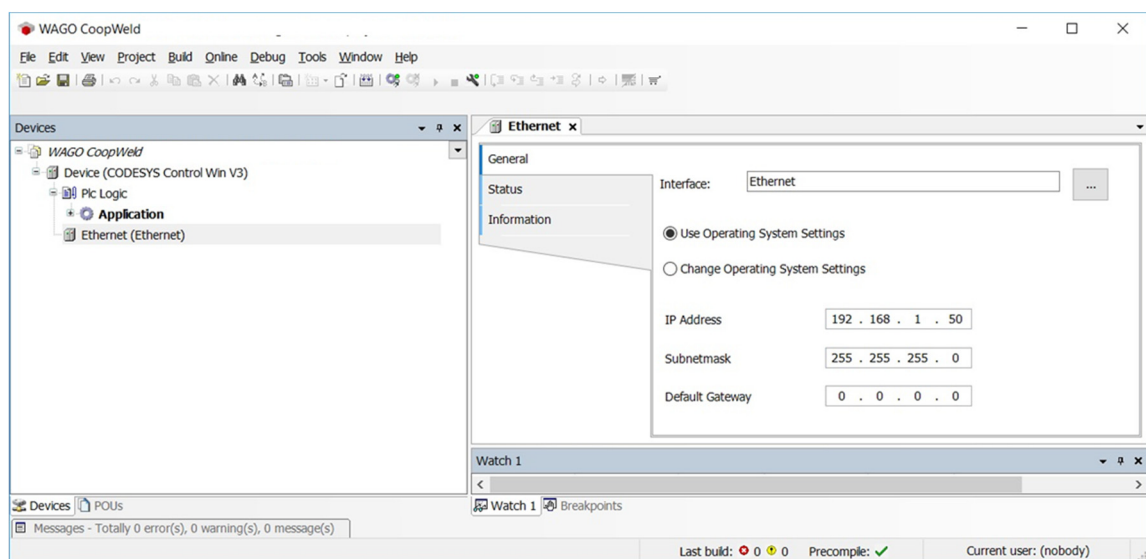


Figura 70 – CodeSys: Adaptador *Ethernet*.

- Sobre o adaptador *Ethernet* adiciona-se o *Scanner*, uma vez que o nosso *softPLC* vai ser o *Scanner* da comunicação e o Módulo de IO's será o *Adapter*.
- Existem fabricantes que fornecem o ficheiro de configuração do equipamento, podendo este ser importado para o CodeSys. Esta opção foi testada, porém, não se obteve comunicação entre os dois equipamentos.

Optou-se, então, por utilizar um *Adapter* genérico já existente no Codesys. Este *Adapter* deve ser subsequente ao *Scanner*.

A

Figura 71 mostra a árvore de dispositivos que se obteve.

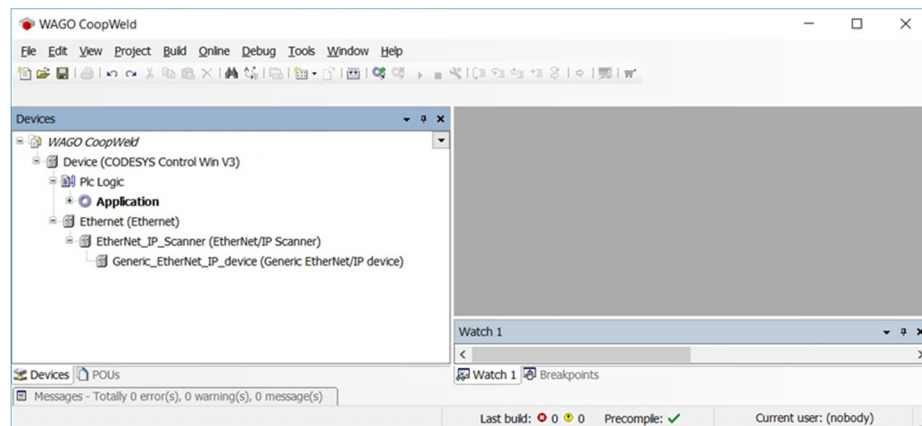


Figura 71 - Codesys: Árvore de dispositivos.

- Na tab do *Generic Adapter*, na subtab *General*, introduz-se o IP correspondente ao módulo de IO's.
- Uma vez que o protocolo a utilizar à Ethernet/IP é necessário especificar as instâncias dos *assemblies*.

Analisando o ficheiro de configuração disponibilizado pelo fabricante [47] é possível depreender as seguintes instâncias:

- *Assembly* da configuração:
 - 1 – *exclusive owner*
 - 2 – *input only*
 - 3 – *listen only*
- *Assembly* dos *outputs*:
 - 65 - *Analog and digital output data*
 - 66 - *Only digital output data*
 - 67 - *Only analog output data*
- *Assembly* dos *inputs*:
 - 68 - *Analog and digital input data plus status*
 - 68 - *Only digital input data plus status*
 - 69 - *Only analog input data plus status*
 - 6B - *Analog and digital input data*
 - 6C - *Only digital input data*
 - 6D - *Only analog input data*

Para o presente trabalho aplica-se:

Assembly da configuração: 1 – *Exclusive owner*

Assembly dos *outputs*: 65 – *Analog and digital output data*

Assembly dos inputs: 6B – Analog and digital input data

Na subtab *Connections* adiciona-se uma nova conexão e preenche-se de acordo com a figura seguinte, Figura 72.

Figura 72 - CodeSys: Configuração *adapter* genérico.

5.2.2 Teste do equipamento controlado pelo CodeSys

Os testes foram realizados com o intuito de avaliar se é possível ter um controlo suficientemente preciso dos atuadores com uma base temporal, para, no programa final, serem usados *timers* com temporizações fixas. Assim, pretende-se obter o valor temporal para cada movimento individual do Virador de modo a definir os *timers* a usar no programa.

Numa primeira fase, foi usado um cronómetro comum para medir os tempos de rotação de cada pá, desde os 0° aos 90° e dos 90° aos 0°, relativamente ao chassi.

Apresentação e análise dos resultados

Os dados apresentados neste subcapítulo foram registados ao longo de uma manhã de testes com o equipamento quase sempre em funcionamento.

Os tempos foram medidos ao longo de 4 rondas igualmente distribuídas ao longo da manhã. Entre as rondas foram realizados outros testes ao nível do programa, mas sempre com os Viradores em movimento.

Dado o carater manual da metodologia utilizada foi repetida, em cada ronda, a medição 4 vezes. A Tabela 25 mostra os tempos medidos, em segundos (s).

Tabela 25 - Tempos de subida e descida da Pá esquerda e da Pá direita

1ª ronda		Pá esquerda				Pá direita			
0° - 90°		8,18	8,24	8,09	8,03	7,96	7,97	8,12	8,03
90° - 0°		5,54	5,17	5,4	5,41	5,43	5,53	5,75	5,56
2ª ronda		Pá esquerda				Pá direita			
0° - 90°		8,08	8,01	7,89	8,1	7,98	8,19	8,02	8,1
90° - 0°		5,38	5,58	5,4	5,61	5,59	5,77	5,78	5,65
3ª ronda		Pá esquerda				Pá direita			
0° - 90°		8,4	8,45	8,54	8,51	8,31	8,14	8,13	8,22
90° - 0°		5,9	5,79	5,89	5,84	5,81	6,01	6,01	5,97
4ª ronda		Pá esquerda				Pá direita			
0° - 90°		8,78	8,82	8,75	8,99	8,77	8,65	8,76	8,79
90° - 0°		6,66	6,53	6,46	6,56	6,43	6,43	6,48	6,51

É de notar que, ao longo das rondas de medição, os tempos medidos foram aumentando gradualmente, atingindo diferenças superiores a 1s entre o valor medido na 1ª e na 4ª ronda. Assim, como a programação está orientada para a utilização recursiva de *timers*, para controlar o ângulo de cada pá, torna-se impossível um controlo preciso.

De seguida, são apresentadas possíveis justificações para as discrepâncias entre os valores medidos:

- com o aumento da temperatura do óleo a viscosidade diminui, tornando-se mais fluido. Como as válvulas direcionais utilizadas são de gaveta e apresentam as típicas folgas, o facto de o óleo perder viscosidade fará aumentar o caudal de fuga.
- outra possibilidade, é o facto de o sistema ser novo. Os testes realizados poderão ter sido realizados em período de “rodagem” do sistema, justificando as variações temporais pela acomodação dos equipamentos.

Conclusões

Uma vez que não se obteve estabilidade nos tempos medidos durante todo o período de funcionamento, a solução passará pelo abandono do controlo recorrendo aos *timers* e implementar uma solução de controlo com recurso à sensorização.

5.3 Sensorização

Neste subcapítulo será descrito o processo sensorização dos Viradores. Assim, são selecionados os tipos de sensores adequados a cada movimento e posicionados no local mais apropriado.

Por fim, é feita uma validação teórica que atesta a conformidade e o bom funcionamento do equipamento.

5.3.1 Tipos de sensores e detetores

De modo a facilitar a escolha dos dispositivos a utilizar, foi realizado um apanhado dos principais sensores e detetores comumente utilizados na indústria.

- Detetores de proximidade:
 - Indutivo
 - Capacitivo
 - Do tipo contacto *reed*
 - Ótico
- Sensores de posição:
 - Ultrassónico
 - Laser
 - De fio
 - *Encoder* linear
 - *Encoder* angular

5.3.2 Configuração

Existem três variáveis que se pretendem controlar: o ângulo da Pá esquerda, o ângulo da Pá direita (relativamente ao chassi) e a posição da Carruagem interior relativamente à Carcaça exterior.

Tendo em mente todos os sensores acima mencionados foram estudadas várias configurações para controlar cada uma destas variáveis.

Ângulo da Pá (válido para ambas as pás)

1. Três sensores indutivos montados numa estrutura. Esta estrutura está fixa ao corpo do atuador ficando os detetores dispostos ao longo da haste, um para detetar a posição de 0° (atuador recuado), outro para 90° (atuador avançado) e outro para uma posição intermédia. Na extremidade da haste está colocado um aplique metálico para facilitar a deteção.

A Figura 73 ilustra esta configuração, onde os três sensores se encontram numerados.

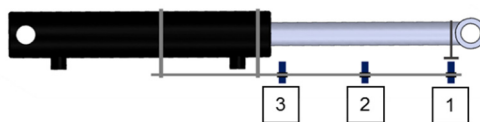


Figura 73 - Sensores - configuração nº 1.

Vantagens

- Adição facilitada de sensores intermédios, ou posições intermédias
- Integração simples com o equipamento de controlo
- Robustez
- Programação acessível

Desvantagens

- Montagem complexa
- Elevado número de componentes (1 sensor para cada posição)
- Espaço necessário
- Número de componentes móveis
- Posicionamento discreto das pás

Em conclusão, é um sistema que pode ser minimalista e económico, mas ser suficiente para o funcionamento pretendido.

2. Esta configuração é semelhante à do ponto 1 com exceção do sensor 3, que é removido e substituído por um detetor do tipo contacto mecânico (fim de curso mecânico) fixo na carruagem, mantendo a função de assinalar que a respetiva pá se encontra a 0°, relativamente ao chassi.

A Figura 74 ilustra esta configuração onde os três sensores se encontram numerados.

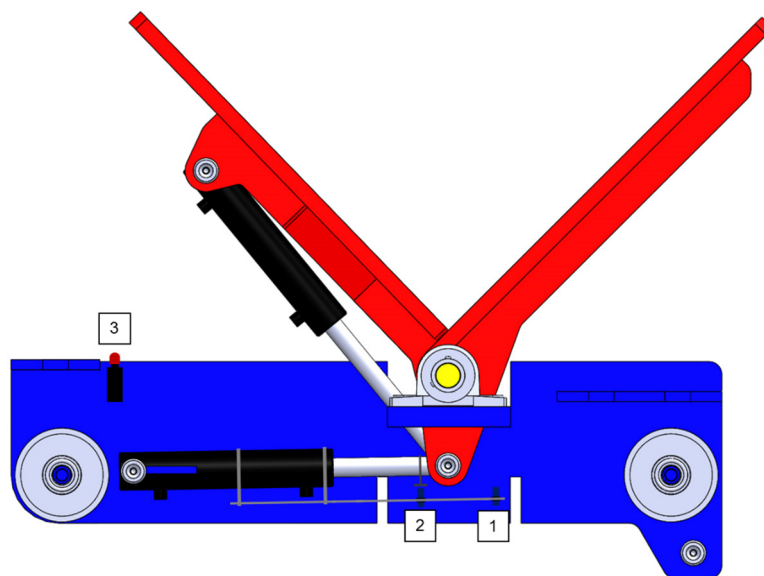


Figura 74 - Sensores - configuração nº 2.

Este sistema, relativamente ao anterior, considera-se mais robusto e económico.

3. No corpo do atuador fixa-se um laser de distância analógico. Na extremidade da haste, coloca-se uma placa perpendicular ao feixe luminoso, para facilitar a detecção; outra opção seria apontar para o veio de amarração do cilindro, embora nesta segunda opção a superfície alvo seja mais pequena.

Na Figura 75 é ilustrada esta configuração onde o feixe luminoso 3, que é emitido pelo sensor 1, tem como alvo a placa refletora 2.

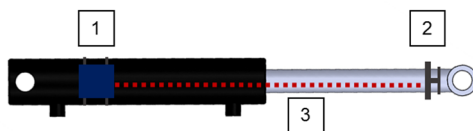


Figura 75 - Sensores - configuração n.º 3.

Vantagens

- Número de componentes reduzido
- Sensor de boa precisão
- Maior nível de controlo (sensor analógico)

Desvantagens

- Dispendioso, relativamente às soluções anteriores
- Programação complexa, relativamente às soluções anteriores
- Menor robustez (equipamento mais sensível ao ambiente, poeiras, etc.)

4. Utilizar um laser de distância analógico fixo ao chassi que pode ter como alvo a face inferior da pá, à esquerda na Figura 76, ou uma face da estrutura inferior da pá, à direita na Figura 76.

Quanto ao tipo de lasers a utilizar para esta aplicação, os mais adequados são os do tipo difuso. Estes funcionam melhor em superfícies rugosas e não perpendiculares ao feixe. Assim, a superfície alvo não precisa de ser refletora e a sujidade da mesma é mais tolerável.

Na Figura 76, à esquerda, o sensor 1 é apontado para a face inferior da Pá esquerda, e à direita, o sensor 1 é apontado para uma face da estrutura da Pá direita.

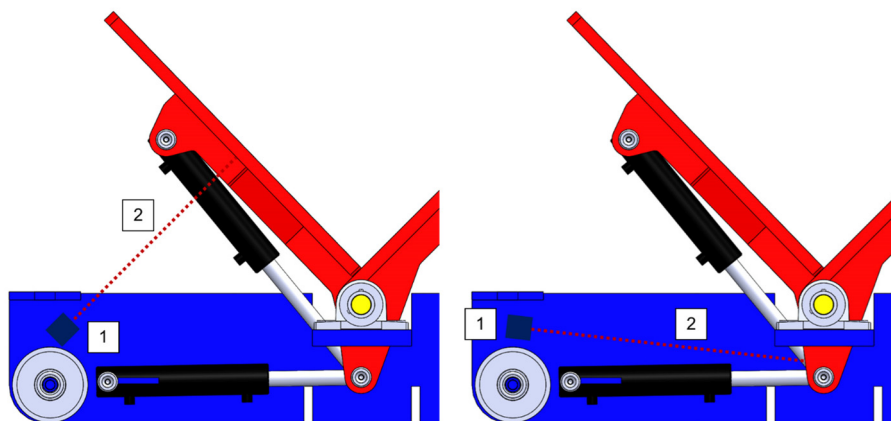


Figura 76 - Sensores - configuração n.º 4.

Vantagens

- As mesmas que na configuração 3
- Menor número de componentes móveis
- Visualmente mais limpo, o sensor ou sensores estão agrupados sob a estrutura do chassi
- Flexibilidade de posicionamento do sensor

Desvantagens

- Semelhantes à configuração 3

Posição da Carruagem interior

A precisão de posicionamento da Carruagem interior é um fator essencial para o bom funcionamento do equipamento, evitando desgaste e esforços desnecessários. Para tal, deverá optar-se por um dos sensores de posição acima listados.

De seguida, é apresentada uma tabela comparativa, Tabela 26, com os possíveis sensores para esta aplicação.

Tabela 26 - Comparação entre os sensores considerados

Sensor	Preço	Precisão	Montagem
Ultrassónico	Baixo	Baixa	Fixo à carcaça, tendo o chassi da carruagem como alvo
Laser Difuso	Alto	Boa	Igual ao ultrassónico
De fio	Médio	Boa	Fixo à carcaça, sendo o chassi da carruagem o ponto de ancoragem
<i>Encoder linear</i>	Alto	Muito boa	Régua fixa à Carcaça exterior e o sensor fixo ao chassi da carruagem

5.3.3 Solução escolhida e aplicação

Com base no que foi apresentado anteriormente, foi elaborada uma lista de opções que, em conjunto com a equipa da SARKKIS Robotics foi analisada e debatida, tendo-se chegado à conclusão que, dado os prazos apertados do projeto, e tendo em conta os prazos de entrega dos equipamentos, a opção de utilizar lasers difusos em todos os atuadores é a mais indicada.

A SARKKIS Robotics, considerando o curto prazo de entrega do fornecedor, disponibilizou seis sensores lasers do tipo difuso: quatro Sick DT20 HI e dois Sick OD2. Apesar de não ser a opção mais económica, é a que permite maior flexibilidade do equipamento, possibilitando, assim, explorar novas possibilidades de funcionamento com um controlo preciso e a malha fechada. Além disso, torna este trabalho mais desafiante.

A Tabela 27 mostra algumas características dos sensores a utilizar.

Tabela 27 - Sensores Sick DT20 HI e SICK OD2 [48] [49]

Sensor	Qtd.	Descrição
SICK DT20 P224B-040405	4	Conector M12 – 5 pinos Output de 4 a 20mA Alcance de medição de 100 a 1000mm
SICK OD2 P250W150I0-6036645	2	Conector M12 – 8 pinos Output de 4 a 20mA Alcance de medição de 100 a 400mm

Controlo dos sensores

Os sensores operam com valores analógicos de 4 a 20mA. Portanto, é necessária a adição de uma ou mais cartas WAGO compatíveis de modo a disponibilizar entradas suficientes para o total dos seis lasers usados no conjunto dos dois Viradores.

As cartas disponibilizadas pela SARKKIS Robotics foram duas WAGO 750-453 para valores analógicos de 0 a 20mA, disponibilizando, cada uma, 4 canais de entrada analógica [50].

O valor recebido pelo *software*, para cada canal, é de 16 *bit*, recebido em dois *bytes* separados, *high byte* e *low byte*, sendo o *high byte* o mais significativo. Foi implementada, na programação do *softPLC* uma função para a converter estes dois *bytes* em uma única *word*.

Através da experimentação percebeu-se que para o valor de 4mA do sensor, o valor recebido pelo *software*, em decimal, é 6520 e, para o valor de 20mA, é recebido 32764. Estes valores são importantes para mais tarde ser feita a conversão dos mesmos para graus, subcapítulo 5.3.4.

Posicionamento

As figuras seguintes, Figura 77 e Figura 78, mostram o posicionamento escolhido para cada um dos lasers.

Utiliza-se um DT20 (Laser 3, Feixe 3, Alvo 3) para o movimento horizontal da carruagem, pois é o único que tem um alcance de medição compatível com o curso de movimento da carruagem. Para a Pá esquerda utiliza-se o segundo DT20 inclinado (Laser 1, Feixe 1), tendo como alvo a face inferior da pá (Alvo 1). Para a Pá direita, utiliza-se o OD2 (Laser 2, Feixe 2) na horizontal, tendo a parte inferior da estrutura da pá como alvo (Alvo 2), devido ao seu alcance mais curto, relativamente ao DT20.

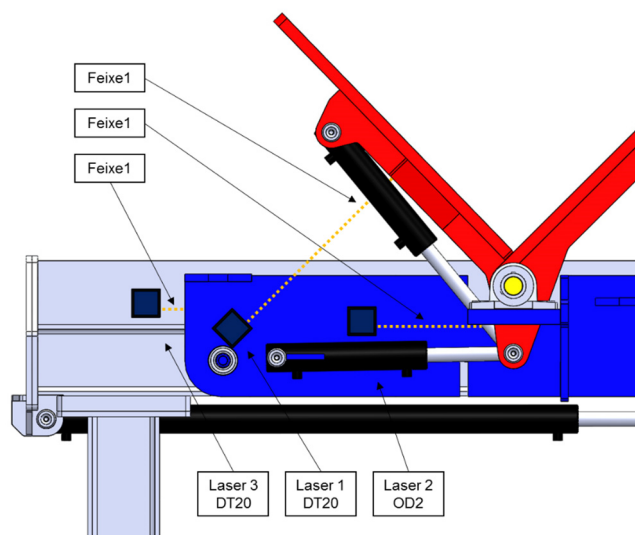


Figura 77- Posicionamento dos sensores, vista 2D.

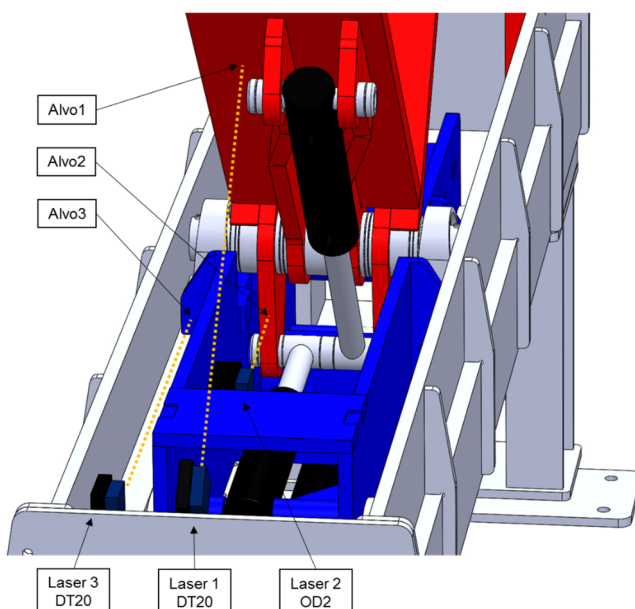


Figura 78 - Posicionamento dos sensores, vista 3D.

Foram, também, adquiridos os cabos correspondentes a cada sensor, com comprimentos compreendidos entre 1,5 e 2m. O somatório de todos os condutores dos 3 cabos, que cada Virador possui, é de 18 condutores.

Uma vez que, os cabos adquiridos não têm comprimento suficiente para se ligarem diretamente aos módulos WAGO, que estão no interior do quadro elétrico, e existem condutores que não estão a ser utilizados ou que podem ser comuns a vários sensores, será necessária uma caixa de bornes.

Esta caixa elétrica fica colocada no Virador, da qual sai apenas um cabo contendo apenas os condutores necessários para o funcionamento pretendido.

Todos os lasers estão aparafusados a um suporte que, por sua vez, é fixo à superfície do Virador por fita adesiva de dupla face extra forte. Estes suportes permitem um pequeno ajuste da posição do sensor e conferem o espaçamento necessário para que o ponto alvo seja o pretendido. No Anexo B é descrito o projeto destes suportes.

5.3.4 Análise matemática

A análise matemática deste subcapítulo pretende demonstrar, analiticamente, como foi conseguida a relação entre:

- Valor medido pelo Laser 1 e o valor do ângulo que a Pá esquerda faz com o chassi.
- Valor medido pelo Laser 2 e o valor do ângulo que a Pá direita faz com o chassi.
- Valor medido pelo Laser 3 e a distância que a Pá esquerda (na posição vertical) faz com os batentes.

Laser 1

Pretende-se a relação entre o valor do laser e o ângulo que a Pá esquerda faz com o chassi.

Porém, esta relação não é direta, mas pode ser conseguida recorrendo a relações trigonométricas. A Figura 79 apresenta o esquema do método que foi utilizado e denomina todas as variáveis utilizadas.

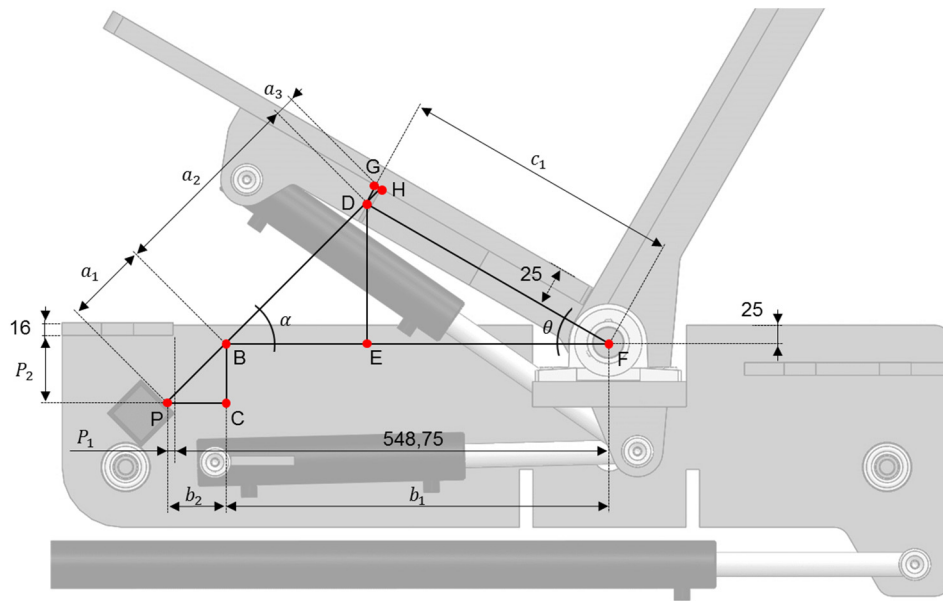


Figura 79 - Laser 1 - Relações trigonométricas.

O valor em mm medido pelo laser, Y , é dado por:

$$Y = a_1 + a_2 + a_3 \quad (14)$$

Contudo, o valor de a_1 é unicamente função do posicionamento do sensor e não de θ . Como o laser vai ser calibrado para a posição da pá a 0° , para os 4mA, e a 90° para 20mA, pode não se incluir o valor de a_1 no somatório anterior.

Assim, pode considerar-se que o valor do laser, função de θ , corresponde a:

$$Y = a_2 + a_3 \quad (15)$$

Adicionalmente, o valor medido pelo laser em mm é a conversão linear do valor recebido pelo *software*, X , em decimal, ou seja:

$$Y = m.X + b \quad (16)$$

Sendo que, m é a constante de proporcionalidade e b é a ordenada na origem.

Relação entre a_2 e θ :

O valor de a_2 é obtido a partir do triângulo BDF, onde:

$$b_1 = a_2 \cdot \cos(\alpha) + c_1 \cdot \cos(\theta) \quad (17)$$

Para o valor de c_1 , continuando no triângulo BDF:

$$\begin{aligned} a_2 \cdot \sin(\alpha) &= c_1 \cdot \sin(\theta) \\ (=) c_1 &= a_2 \cdot \frac{\sin(\alpha)}{\sin(\theta)} \end{aligned} \quad (18)$$

Quanto ao valor de b_2 sabe-se que:

$$b_2 = 548,75 + P_1 - b_1 \quad (19)$$

e,

$$b_2 = \frac{P_2 + 16 - 25}{\tan(\alpha)} \quad (20)$$

Logo, b_1 fica:

$$b_1 = 548,75 + P_1 - \frac{P_2 - 9}{\tan(\alpha)} \quad (21)$$

Substituindo b_1 e c_1 lá em cima, fica:

$$\begin{aligned} 548,75 + P_1 - \frac{P_2 - 9}{\tan(\alpha)} &= a_2 \cdot \cos(\alpha) + a_2 \cdot \frac{\sin(\alpha)}{\sin(\theta)} \cdot \cos(\theta) \\ (=) a_2 \cdot \left(\cos(\alpha) + \frac{\sin(\alpha)}{\tan(\theta)} \right) &= 548,75 + P_1 - \frac{P_2 - 9}{\tan(\alpha)} \end{aligned} \quad (22)$$

A relação entre a_2 e θ fica definida quando se escolher a posição e a inclinação do laser, variáveis P_2 e α .

Relação entre a_3 e θ :

A Figura 80 é a reprodução ampliada do triângulo DGH da Figura 79.

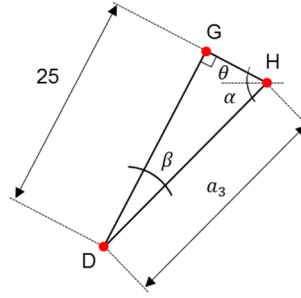


Figura 80 - Laser 1 - Triângulo DGH.

Deste triângulo tira-se que:

$$\beta = 90 - (\alpha + \theta) \quad (23)$$

Logo, a_3 :

$$a_3 = \frac{25}{\cos(90 - (\alpha + \theta))} \quad (24)$$

A relação entre a_3 e θ fica definida quando se escolher a inclinação do laser, α .

Relação entre Y e θ :

A relação entre o valor, em mm, medido pelo laser, Y , e o ângulo da Pá esquerda, θ , fica definida quando se posicionar o laser, variáveis P_1 , P_2 e α , através da equação:

$$Y = \frac{548.75 + P_1 - \frac{P_2 - 9}{\tan(\alpha)}}{\cos(\alpha) + \frac{\sin(\alpha)}{\tan(\theta)}} + \frac{25}{\cos(90 - (\alpha + \theta))} \quad (25)$$

Relação entre X e θ :

Para se definir esta relação basta encontrar o valor das constantes m e b .

$$m = \frac{Y_{@ \theta=90^\circ} - Y_{@ \theta=0^\circ}}{32764 - 6520} \quad (26)$$

$$b = Y_{@ \theta=90^\circ} - m \cdot 32764 \quad (27)$$

Os valores de P_1 e P_2 foram atribuídos experimentalmente de forma a colocar o sensor num local favorável, sendo estes de 42,6 e 89,0mm, respetivamente.

Assim, o gráfico da Figura 81 apresenta a curva que relaciona este valor com o ângulo que a pá faz com o chassi, para diferentes inclinações do sensor, α , mantendo constantes os valores de P_1 e P_2 .

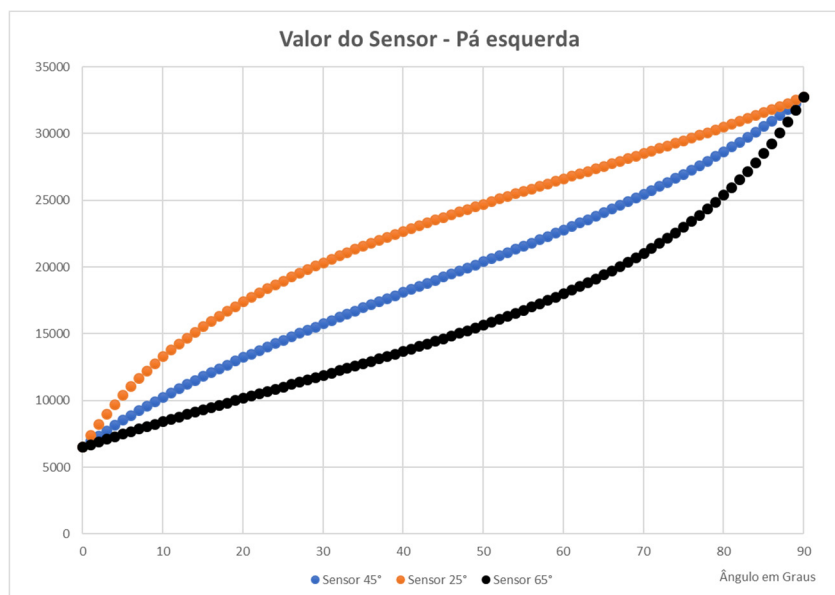


Figura 81 - Laser 1 - Gráfico do valor do sensor em função do ângulo da pá para várias inclinações do Laser 1.

Analisando o movimento da pá entre os 0° e os 90°, é possível perceber que se a velocidade de avanço do cilindro for constante, para inclinações do laser inferiores a 45°, a taxa de variação do ângulo da pá tende a ser menor no início da trajetória. Por outro lado, para inclinações superiores a 45°, a taxa de variação do ângulo da pá tende a ser menor no final da trajetória.

Com inclinações de 45°, a taxa de variação do ângulo da pá não é constante, apresentando-se como um caso intermédio entre os dois casos acima referidos. Contudo, tende a ser menos pronunciada. Tende a ser inferior, em módulo, no início e no final da trajetória. À medida que se aumenta o ângulo de inclinação do laser, a sua taxa de variação tende a tomar um valor constante no início da trajetória e tende a ficar mais pronunciada no final. Com a diminuição do ângulo de inclinação, a taxa de variação do ângulo tende a ser constante no final da trajetória e inferior no início.

O posicionamento do sensor com uma inclinação de 45° é aquele que apresenta menores variações da taxa de variação do ângulo. Logo, é o posicionamento que apresenta o comportamento mais próximo do linear, sendo por isso o escolhido.

Para conseguir fazer o controlo, no *softPLC*, do avanço e do recuo do atuador, partindo do valor do ângulo da pá, resta reorganizar a equação em ordem a θ .

Para a reorganização da equação em ordem a θ , foram usadas duas ferramentas matemáticas, o MATLAB e a ferramenta online WolframAlpha. Porém, o resultado obtido foi demasiado complexo e extenso para ser fácil e rapidamente processado pelo *softPLC*.

Optou-se, assim, por fazer uma aproximação polinomial utilizando o Microsoft Excel, Figura 82.

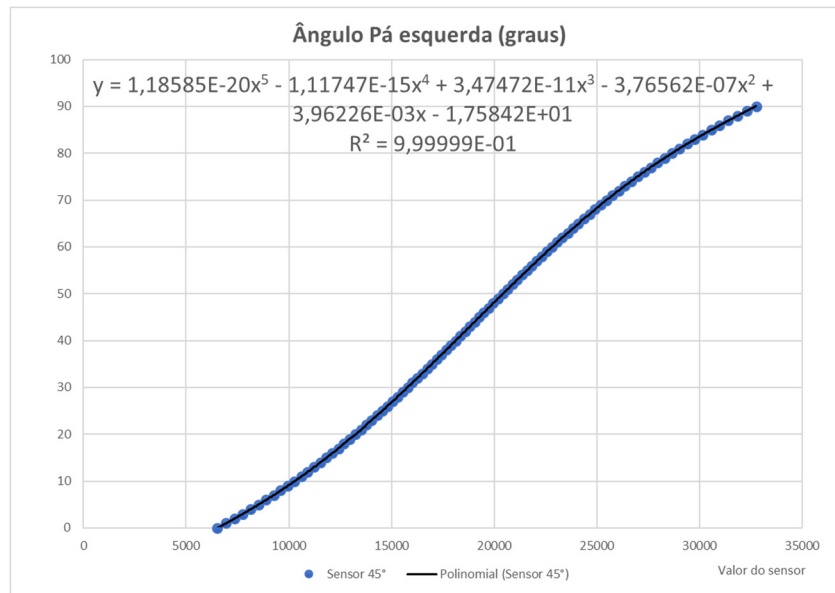


Figura 82 - Laser 1 - Aproximação polinomial do ângulo da pá em função do valor do sensor recebido.

Laser 2

Este laser funciona de forma semelhante ao Laser 1, variando o seu valor conforme o ângulo que a Pá direita faz com o chassi, entre 0° a 90°.

Mais uma vez, esta relação não é direta, mas pode ser obtida através de relações trigonométricas. A Figura 83 apresenta o esquema do método que foi utilizado e denomina todas as variáveis necessárias.

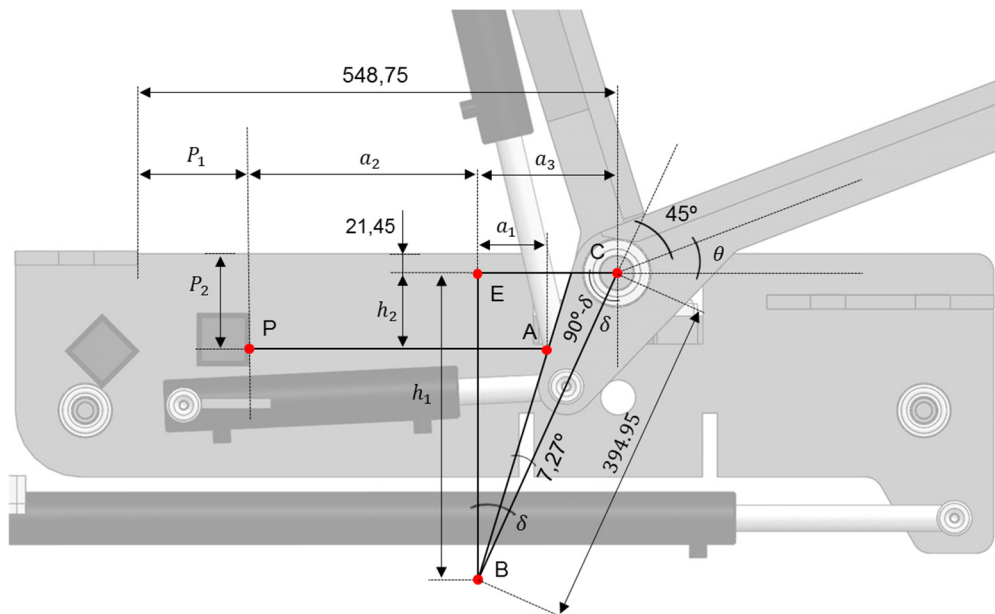


Figura 83 - Laser 2 - Relações trigonométricas.

O valor, em mm, medido pelo laser, Y , é dado por:

$$Y = a_1 + a_2 \quad (28)$$

Sendo X , o valor do laser, em decimal, recebido pelo *software*:

$$Y = m.X + b$$

Relação entre δ e θ :

$$\delta = 45 - \theta \quad (29)$$

Relação entre a_1 e δ :

$$a_1 = \overline{BA} \cdot \text{sen}(\delta - 7,27) \quad (30)$$

Sendo:

$$\overline{BA} = \frac{h_1 - h_2}{\cos(\delta - 7,27)} \quad (31)$$

$$h_1 = 394,95 \cdot \cos(\delta) \quad (32)$$

$$h_2 = P_2 - 21,45 \quad (33)$$

Fica:

$$a_1 = (394,95 \cdot \cos(\delta) - P_2 + 21,45) \cdot \tan(\delta - 7,27) \quad (34)$$

Relação entre a_2 e δ :

$$a_2 = 548,75 - P_1 - a_3 \quad (35)$$

Como:

$$a_3 = 394,95 \cdot \text{sen}(\delta) \quad (36)$$

Fica:

$$a_2 = 548,75 - P_1 - 394,95 \cdot \text{sen}(\delta) \quad (37)$$

Relação entre Y e δ :

A relação entre o valor, em mm, medido pelo laser, Y , e o ângulo, θ , da Pá esquerda, fica definido quando se posicionar o laser, variáveis P_1 , P_2 e δ , através da equação:

$$Y = (394,95 \cdot \cos(\delta) - P_2 + 21,45) \cdot \tan(\delta - 7,27) + 548,75 - P_1 - 394,95 \cdot \text{sen}(\delta) \quad (38)$$

Relação entre Y e θ :

$$Y = (394,95 \cdot \cos(45 - \theta) - P_2 + 21,45) \cdot \tan(45 - \theta - 7,27) + 548,75 - P_1 - 394,95 \cdot \text{sen}(45 - \theta) \quad (39)$$

Relação entre X e θ :

Para se definir esta relação basta encontrar o valor das constantes m e b .

$$m = \frac{Y_{@ \theta=90^\circ} - Y_{@ \theta=0^\circ}}{32764 - 6520} \quad (40)$$

$$b = Y_{\theta=90^\circ} - m \cdot 32764 \quad (41)$$

Ficando agora a relação entre X e θ totalmente definida quando se seleccionar os valores de P_1 e P_2 .

Os valores de P_1 e P_2 foram atribuídos experimentalmente de forma a colocar o sensor num local favorável, sendo estes de 292,5 e 70,0mm, respetivamente.

Tal como foi feito para o Laser 1, e devido às mesmas razões, utilizou-se o Microsoft Excel para fazer uma aproximação polinomial a fim de se reorganizar a equação em ordem a θ .

Assim, o gráfico da Figura 84 apresenta a curva e a respetiva regressão polinomial, que rege a relação entre θ e X .

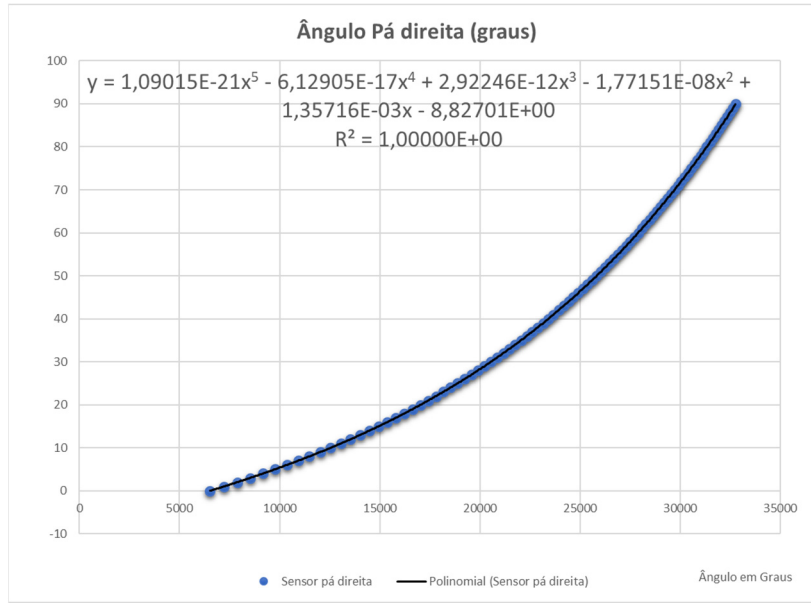


Figura 84 - Laser 2 - Aproximação polinomial do ângulo da pá em função do valor do sensor recebido.

Laser 3

A relação entre o valor do laser, em decimal, e a posição da Carruagem interior é facilmente obtida, pois trata-se de uma relação linear.

Sendo: X o valor do laser, em decimal, recebido no *software*; C_{min} e $C_{máx}$ os valores mínimos e máximos da distância da Carruagem interior aos batentes; Y o valor da distância entre a carruagem e os batentes, temos:

$$Y = m \cdot X + b$$

Onde,

$$m = \frac{C_{min} - C_{máx}}{32764 - 6520} \quad (42)$$

$$b = C_{máx} - m \cdot 6520 \quad (43)$$

Experimentalmente mediu-se o valor de C_{min} e $C_{máx}$:

$$C_{min} = 70$$

$$C_{máx} = 880$$

5.3.5 Validação com dispositivo real

Neste capítulo é feita a comparação entre os ângulos medidos pelos lasers 1 e 2, utilizando as equações acima conseguidas, e o ângulo real que cada pá faz com o chassi. É feita, também, a validação da equação conseguida para o valor da distância da carruagem aos batentes, comparando-a com a distância real.

A medição dos ângulos foi realizada manualmente utilizando um aristo e outras réguas de desenho convencionais. Para a medição da posição da carruagem foi utilizada uma fita métrica. Todos estes dispositivos de medição, apesar da fraca resolução, são suficientes para validar as equações.

Lasers 1 e 2

A Tabela 28 mostra os resultados das medições teóricas e reais do ângulo entre cada uma das pás e o chassi. Devido à fraca resolução dos equipamentos de medição, na coluna dos valores reais foram utilizadas setas indicativas da tendência do valor medido, ou seja, o valor de 9,0↓ significa que o valor do ângulo medido se encontra entre 8,5 e 9,0mm.

Tendo em conta o equipamento e o método de medição utilizados, a diferença entre os valores teóricos e reais apresenta, para a Pá esquerda, um desvio máximo de 1,41134° e desvio padrão de 0,528389° e, para a Pá direita, um desvio máximo de 0,44038° e desvio padrão de 0,553241°. Assim, confirma-se que as equações teóricas conseguidas no subcapítulo 5.3.4 descrevem corretamente o funcionamento real das pás do Virador.

Tabela 28 – Comparação dos ângulos teóricos e reais das pás do Virador, valores em [graus]

Pá esquerda			Pá direita		
Valor Laser	Ângulo Teórico	Ângulo Real	Valor Laser	Ângulo Teórico	Ângulo Real
6520	-0,006960	0,0	6520	-0,019310	0,0
7928	3,431822	4,0	11784	8,553846	9,0↓
9544	7,805971	7,5↑	13384	11,671960	12,0
10784	11,546250	11,5↓	15424	16,097800	16,0
11896	15,203430	15,0	17364	20,846750	21,0↑
13240	19,997560	20,0↓	19396	26,474910	27,0↓
14776	25,928220	25,0↑	21204	32,129650	32,5
16588	33,411340	32,0	22492	36,578920	37,0
17892	39,007890	38,0	23204	39,203470	41,0
19308	45,163910	45,0	24764	45,402850	45,0↓
20308	49,495070	49,0	25372	47,998550	49,0
21204	53,323480	52,5↑	26712	54,107980	54,0↑
22060	56,906200	55,5↑	27628	58,615380	60,0↓
23604	63,106640	62,0	28724	64,393530	65,0↑
25044	68,498380	68,0	29372	68,021110	69,0↓
26036	71,955330	71,0↓	30380	73,999030	75,0↓
27004	75,112000	74,0↑	31052	78,224920	79,0↓
28828	80,479320	80,0↓	31564	81,581310	82,0↓
30932	85,852790	86,0↓	32060	84,950510	85,5
32612	89,766560	89,5	32764	89,940380	89,5
32764	90,116240	90,0↓	Desvio máximo:		0,44038
Desvio máximo:		1,41134	Desvio padrão:		0,553241
Desvio padrão:		0,528389			

Laser 3

A Tabela 29 mostra os valores das medições teóricas e reais da posição da carruagem, tendo-se concluído que a aproximação teórica conseguida no subcapítulo 5.3.4 descreve com sucesso o posicionamento real da carruagem do Virador.

Tabela 29 - Comparação da posição teórica e real da posição da carruagem, valores em [mm]

Posição da carruagem		
Valor Laser	Posição Teórico	Posição Real
6520	880,00	880,0
12496	695,56	695,0
20644	443,83	444,0
32764	70,00	70,0
Desvio máximo:		0,56
Desvio padrão:		0,275

O desvio máximo observado no posicionamento da carruagem foi de 0.56mm com desvio padrão de 0.275mm. Conclui-se que a aproximação feita para o Laser 3 descreve o comportamento real do posicionamento da Carruagem interior.

5.4 Programação

Para o controlo dos Viradores, será utilizado o *softPLC* da 3S-Smart e utilizando o CodeSys como ambiente de desenvolvimento para a sua programação [46].

A programação é feita com base num programa principal, Programa *Main*, que faz a monitorização do estado em que a célula se encontra e, em função disso, instancia blocos de funções que encapsulam funcionalidades do sistema.

De forma a facilitar a análise de todo o programa de controlo dos Viradores, foram elaborados graficets que visam descrever apenas o funcionamento base do programa, pois não representam exatamente o programa real, não contemplam todas as suas funcionalidades e não foram concebidos tendo em consideração a forma como os PLC correm os programas.

A totalidade do programa foi escrito em texto estruturado segundo na norma IEC 61131-3 e encontra-se no Anexo C desta dissertação.

No capítulo 6 são apresentadas funcionalidades adicionais, que foram implementadas no programa.

De seguida, são listados todos os graficets que são utilizados.

- G1 – Programa *Main*
- G2 – Modo Manual
- G3 – Modo Automático (FB)
- G21 – Movimentos elementares (FB)
- G22 – Operações elementares (FB)

Uma vez que, os blocos de funções dos movimentos e operações elementares são usados recursivamente, são detalhados de seguida. Posteriormente, são detalhados os outros estados de funcionamento.

5.4.1 Movimentos elementares - bloco de funções

Cada movimento elementar corresponde ao acionamento de um solenoide (de cada Virador) que se traduz no avanço ou recue de um cilindro que, por sua vez, permite realizar o movimento elementar pretendido.

O conjunto de movimentos elementares foi programado em texto estruturado e encapsulado por um bloco de funções. A Figura 85 retrata, graficamente, o bloco de funções utilizado.

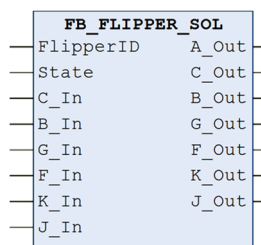


Figura 85 - Programação Viradores: Bloco de funções dos movimentos elementares.

O funcionamento deste bloco de funções é descrito pelo grafcet parcial G21 da Figura 86.

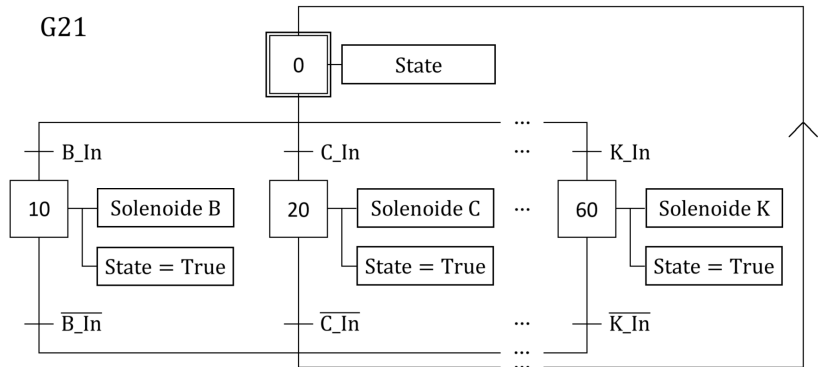


Figura 86 - Programação Viradores: G21 – Movimentos elementares.

Cada uma das etapas tem como ação a energização de um solenoide.
De seguida, são listados os solenoides disponíveis e a respetiva função.

- Solenoide B – Recuo do Atuador 1
- Solenoide C – Avanço do Atuador 1
- Solenoide F – Recuo do Atuador 2
- Solenoide G – Avanço do Atuador 2
- Solenoide J – Recuo do Atuador 3
- Solenoide K – Avanço do Atuador 3

5.4.2 Operações elementares - bloco de funções

Uma operação elementar corresponde ao encadeamento de vários movimentos elementares. Todas as operações foram programadas em texto estruturado e encapsulado por um bloco de funções. A Figura 87 retrata, graficamente, o bloco de funções utilizado.

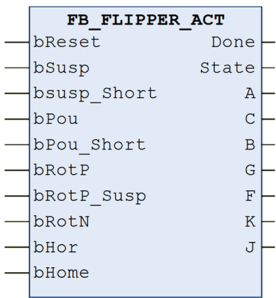


Figura 87 - Programação Viradores: Bloco de funções das operações elementares.

O funcionamento deste bloco de funções é descrito pelo grafcet parcial G22 da Figura 88.

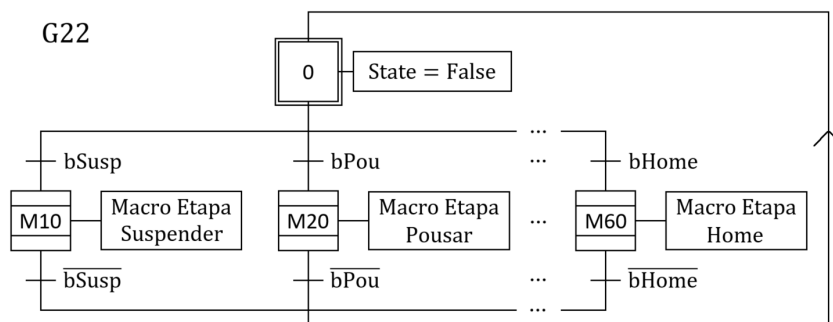


Figura 88 - Programação Viradores: G22 – Operações elementares.

No grafcet parcial G22, cada macro etapa corresponde a uma operação elementar. De seguida, na Figura 89, é demonstrado o funcionamento da operação de suspender a viga. As outras operações elementares desenrolam-se de forma semelhante.

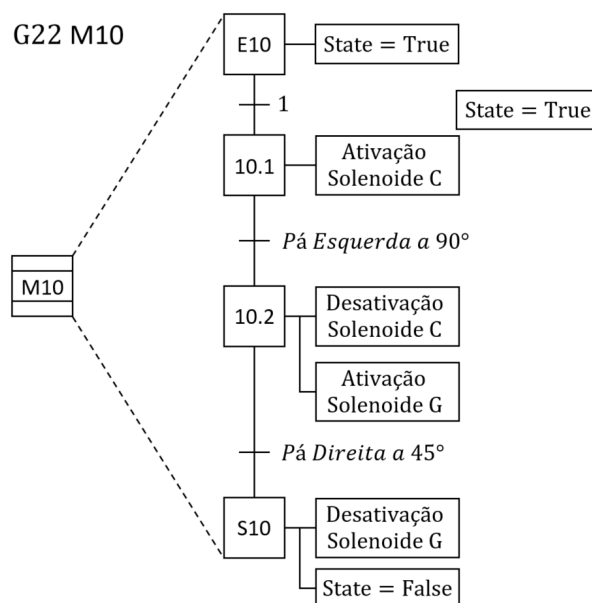


Figura 89 - Programação Viradores: Macro etapa M10 do G22.

As operações elementares ou macro etapas criadas são: Suspende, Pousar, Rotação no sentido Positivo, Rotação no sentido Negativo, Movimento Horizontal e *Home*. Foi implementada, também, a possibilidade de executar apenas parte de uma macro etapa, variáveis bsusp_Short e bpou_Short (Figura 87).

Para a realização da operação elementar de movimentação horizontal da viga, o programa do *softPLC* está preparado para receber do *software* SARKKIS Robotics, por ModBus, informações relativas às dimensões horizontal e vertical da viga.

5.4.3 Programa Main

A célula pode estar em modo automático, em modo manual ou em emergência. Este programa principal tem como objetivo monitorizar o estado atual da célula e, em função deste, instancia blocos de funções que habilitam certas funcionalidades.

O Grafcet 1 (G1 – Programa *Main*), representado pela Figura 90, demonstra como é feita a gestão do estado de funcionamento.

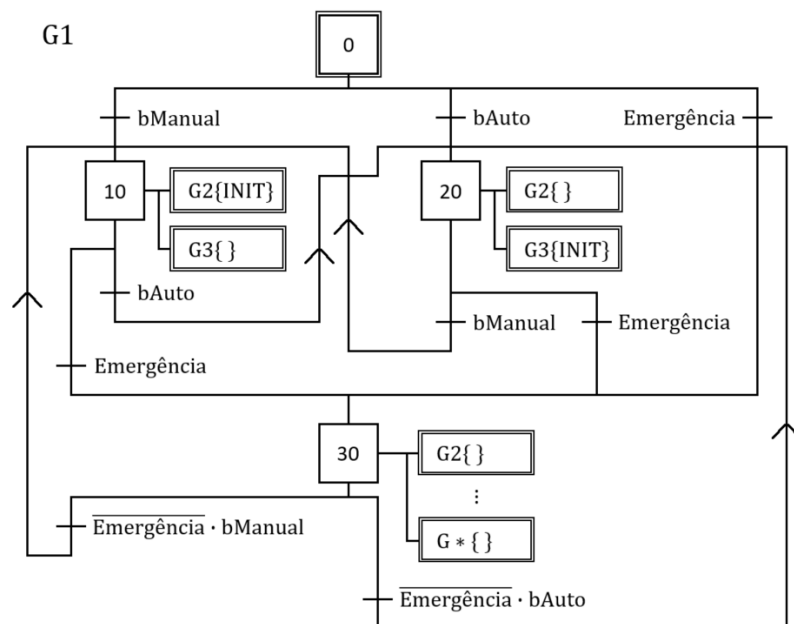


Figura 90 - Programação Viradores: G1 – Programa *Main*.

As variáveis Manual e Auto correspondem a um *input* recebido pelo módulo WAGO que traduz o estado do seletor de modo, situado à entrada da célula. Mediante o estado deste seletor, o grafcet evolui para a etapa 10 ou para a etapa 20.

A etapa 10 corresponde ao modo manual e tem como ações a inicialização do Grafcet 2 e leva os Grafkets 3 e 4 à situação de vazios.

A etapa 20 corresponde ao modo automático e tem como ações a inicialização do Grafcet 3 e leva os Grafkets 2 e 4 à situação de vazios.

Se, em qualquer uma das etapas 0, 10 ou 20, for acionada a variável bEmergência, correspondente ao estado de emergência da célula, que é controlado pelo PLC de segurança, o Grafcet 1 evolui para a etapa 30.

A etapa 30 corresponde ao estado de emergência e tem como ações a inicialização do Grafcet 4 e leva os Grafkets 3 e 4 à situação de vazios.

Nesta secção do programa são apresentadas outras funcionalidades implementadas.

- Conversão do valor do laser num valor decimal utilizável
- Atribuição dos valores dos lasers a cada bloco de funções que os requeiram
- Atualização de variáveis para sinalização do estado da célula (por exemplo, leds da interface gráfica)

5.4.4 Modo manual

Neste modo de funcionamento, é dado ao operador o controlo dos Viradores.

É nesta secção do programa que são instanciados os blocos de funções dos movimentos e das operações elementares.

A Figura 91 apresenta o grafcet parcial G2, que demonstra o funcionamento deste modo de funcionamento.

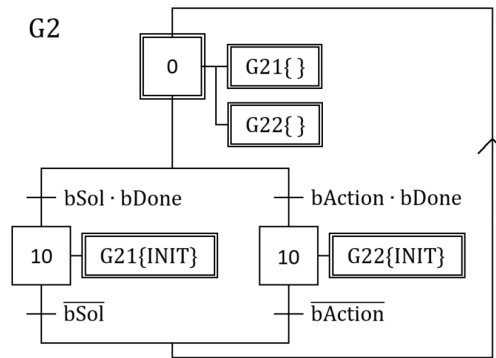


Figura 91 - Programação Viradores: G2 – Modo Manual.

A variável *bDone* é uma variável booleana e significa que o Virador não se encontra em movimento. Assim que se entra na etapa 10 ou 20, esta variável toma o valor *false*.

A variável *bSol* é uma variável booleana que toma o valor *true* assim que o utilizador seleciona um movimento elementar. A variável *bAction* funciona da mesma forma, tomando o valor *true* assim que o operador seleciona uma operação elementar.

Iniciando o funcionamento na etapa 0 e estando os Viradores parados (*bDone* a *true*), se o operador selecionar um movimento elementar, o grafcet evolui para o estado 10. Da mesma forma, se o utilizador selecionar uma operação elementar, o grafcet evolui para a etapa 20.

A etapa 10 inicia o grafcet parcial G21 que demonstra o funcionamento do bloco de funções da movimentação elementar dos Viradores. A etapa 20 inicia o grafcet parcial G22 que demonstra o funcionamento do bloco de funções das operações elementares.

5.4.5 Modo automático - bloco de funções

Ao contrário dos outros modos, em que o utilizador pode movimentar os dois Viradores independentemente um do outro (através da interface gráfica apresentada no capítulo 6), no modo automático os dois Viradores são utilizados sempre em simultâneo. Assim, a programação deste modo garante o sincronismo das várias operações elementares.

O modo automático é, fundamentalmente, o encadeamento de várias operações e movimentos elementares. Mais uma vez, a programação do modo automático foi feita em texto estruturado e foi encapsulada por um bloco de funções. A Figura 92 ilustra o bloco de funções utilizado.

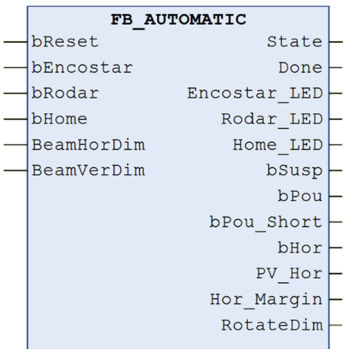


Figura 92 - Programação Viradores: Bloco de funções do modo automático.

Neste modo automático as variáveis de *input* (à esquerda na Figura 92), podem ser controladas pelo operador, através da interface gráfica ou pelo *software* da SARKKIS Robotics através de ModBus.

O grafcet parcial G3, Figura 93, descreve o funcionamento do modo automático e, a Figura 94, descreve o princípio de funcionamento da macro etapa M10 deste mesmo grafcet.

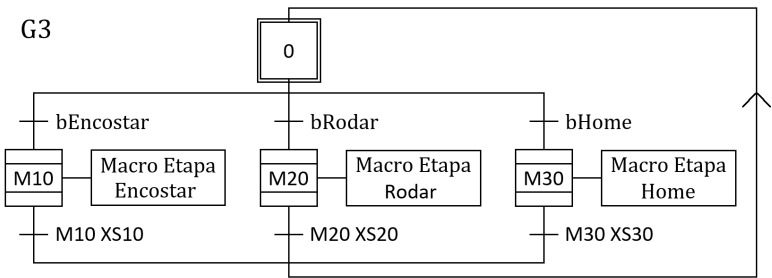


Figura 93 - Programação Viradores: G3 – Modo Automático.

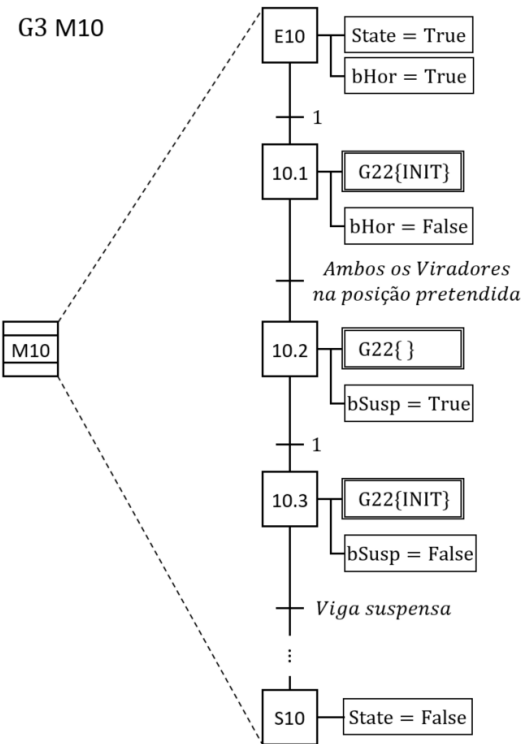


Figura 94 - Programação Viradores: Macro etapa M10 do G3.

Nesta parte do programa, foi implementado o registo das operações realizadas. Assim, sempre que se pretende executar uma nova operação o programa tem em memória qual foi última operação executada, evitando, assim, executar operações repetidas incompatíveis, como é o caso da operação de Encostar. Este registo é apagado sempre que a célula entra em modo manual ou em emergência.

A macro etapa Rodar tem uma duração de 30s, as outras macro etapas não têm duração fixa pois esta depende da posição inicial da viga e da sua dimensão.

De seguida, é descrito o funcionamento de cada uma das macro etapas desenvolvidas, listando os vários passos intermédios e considerações tomadas.

- Macro etapa Encostar
 - A viga deve estar pousada sobre a Pá direita
 - Operação Suspend
 - Movimento horizontal da viga até à proximidade do batente
 - Pousa a viga utilizando a macro etapa iniciada pela variável bpou_Short onde se pousa a viga baixando apenas a Pá direita, ficando a Pá esquerda a 90° com o chassi.
 - A viga fica a 5mm dos batentes
 - Por arrastamento da viga e feito o encosto da mesma aos batentes
 - Movimento horizontal das pás 5mm para a esquerda, margem de segurança
 - Operação Pousar
 - Movimento horizontal para a esquerda para garantir que as pás ficam completamente no interior chassi

- Macro etapa Rodar
 - A operação de Rodar só pode ser executada seguidamente à operação de Encostar ou Rodar
 - Movimento horizontal até que a distância da pá esquerda (na vertical) à face da viga seja de 5mm
 - A posição em que a viga foi deixada pela etapa anterior foi memorizada
 - Operação Suspend
 - Movimento horizontal para a direita de modo a ter espaço suficiente à esquerda da viga para a poder rodar 90°
 - Operação Rotação no sentido Positivo
 - Movimento horizontal para a esquerda de modo a colocar o eixo de rotação das pás à esquerda da viga, deste modo a operação automática de Rodar ou Encostar pode ser executada novamente

- Macro etapa *Home*
 - Pousa apenas a Pá direita (bPou_Short)
 - Se esta não atingir a posição pretendida em 15s é disparada uma mensagem de erro e a operação é cancelada
 - Operação Pousar
 - Se a Pá esquerda não atingir a posição pretendida em 15s é disparada uma mensagem de erro e a operação é cancelada
 - Movimento Horizontal da Carruagem interior para a esquerda
 - Se esta não atingir a posição pretendida em 40s é disparada uma mensagem de erro e a operação é cancelada

- Se cada movimento for executado dentro do tempo normal, no final de cada um deles é disparada uma mensagem de bom funcionamento
- Estas mensagens aparecem na interface gráfica da Figura 95

5.4.6 Emergência

Quando a célula entra em emergência, Etapa 30 do grafcet G1 representado pela Figura 90, todos os blocos de funções são parados através de uma variável de *reset* existente em cada um deles. Assim, se algum dos Viradores estiver em movimento este é cessado, sendo atingido um estado seguro.

6 Interface Gráfica - SoftPLC

Durante a realização desta dissertação foi sendo desenvolvida uma interface gráfica que serviu de suporte para todo o desenvolvimento e todos os testes realizados com os Viradores.

Esta interface gráfica foi concebida utilizando o CodeSys e é ilustrada pela Figura 95.

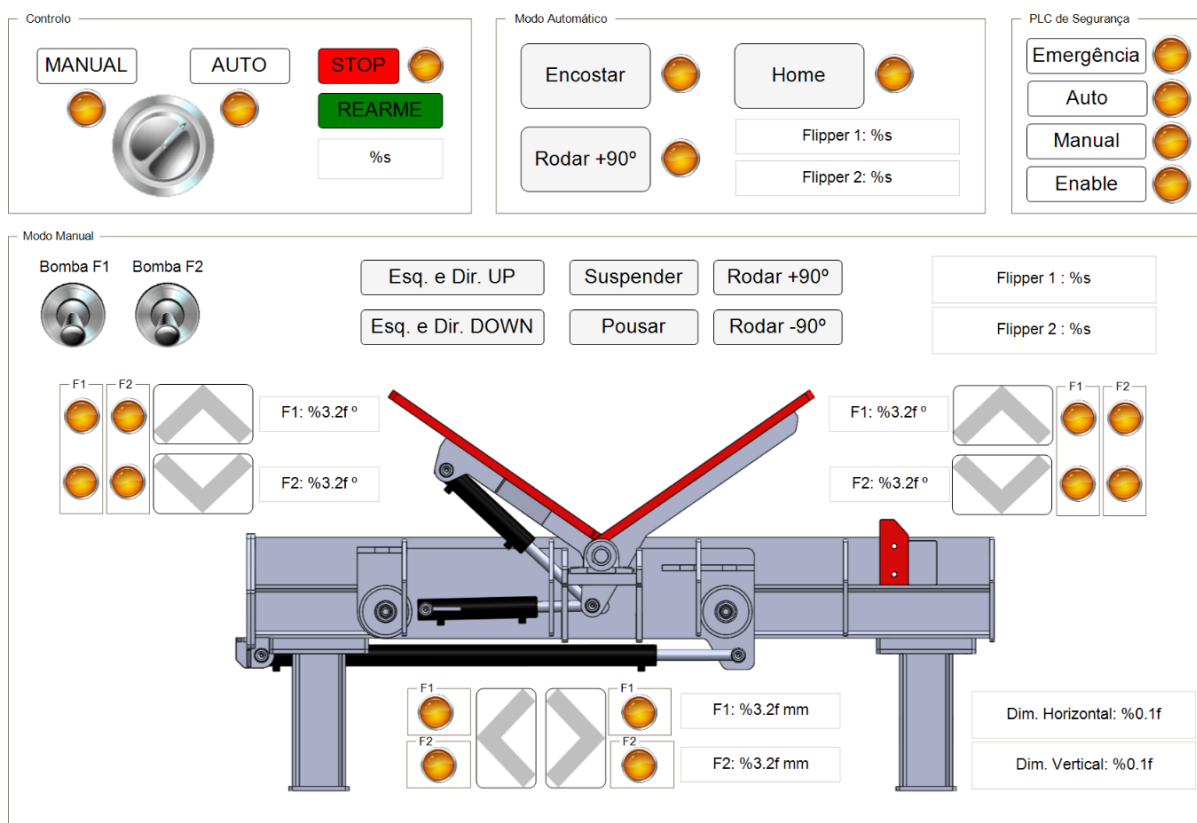


Figura 95 - Interface gráfica: *Vizualisation* do CodeSys.

Esta interface pode ser dividida em 4 secções (GroupBoxs da Figura 95): Controlo, PLC de segurança, Modo Automático e Modo Manual.

GroupBox Controlo

Esta secção simula o seletor de modo do exterior da célula e o conjunto da botoneira de emergência e respetivo botão de rearme. Existe, também, uma caixa de texto que informa qual o estado do sistema (Manual, Auto ou Emergência).

GroupBox PLC de segurança

Nesta secção o utilizador apenas é informado acerca do estado real do PLC de segurança.

GroupBox Modo Automático

Nesta secção é permitido ao operador executar operações implementadas no controlo automático da célula, simulando, assim, o controlo de variáveis que seriam controladas do software SARKKIS Robotics.

As duas caixas de texto permitem ao utilizador saber qual a operação que cada Virador está a realizar, por exemplo: “Rotação Positiva”, “Movimento horizontal esquerda”, entre outros.

GroupBox Modo Manual

Esta secção permite o controlo total dos Viradores para o operador.

À esquerda, existem dois *toggle buttons* responsáveis por ligar a unidade hidráulica de cada Virador independentemente uma da outra, permitindo, assim, movimentar cada Virador de forma independente.

Em cima e ao centro existem 6 botões que iniciam operações elementares e, nas caixas de texto à direita dos botões, o utilizador é informado acerca do movimento que cada Virador está a executar, por exemplo: “Deslocar para a esquerda”, “Pá esquerda de 0° até 90°”, entre outros.

No centro da GroupBox está uma imagem do Virador onde os componentes internos são visíveis. Ao lado de cada componente móvel, Pá direita, Pá esquerda e Carruagem interior, encontram-se os respetivos botões de controlo. Adicionalmente, é mostrado o posicionamento de cada um destes componentes, para cada um dos Viradores.

No canto inferior direito, o operador é informado acerca da dimensão horizontal e vertical da viga em cada momento. Estes dois valores são comutados á medida que a viga vai sendo rodada.

Contudo, e como referido na introdução desta dissertação, é um objetivo a conceção de uma interface gráfica para o controlo manual dos Viradores que possa ser integrada no *software* SARKKIS Robotics.

Assim, foi desenvolvida, em C# e WPF, a interface gráfica da Figura 96.

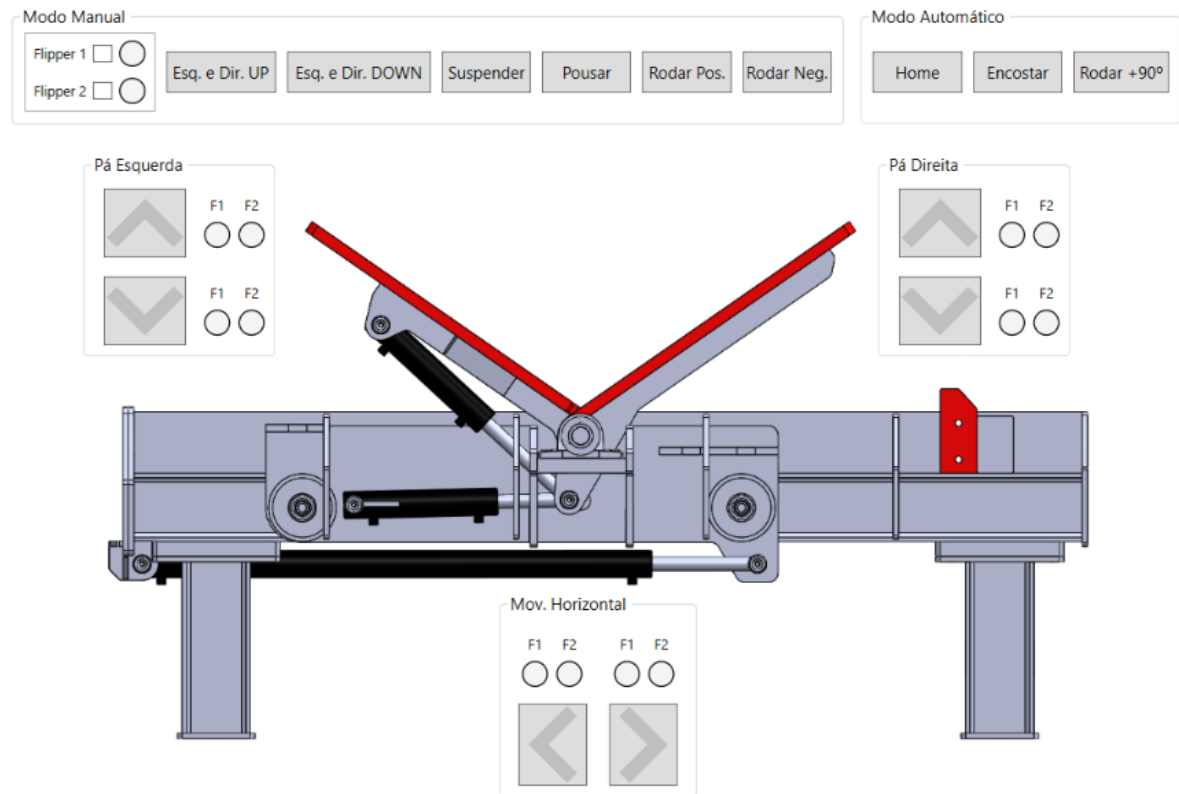


Figura 96 - Interface gráfica: WPF.

Esta interface gráfica foi desenvolvida para ser utilizada na HMI da célula e não em ambiente de testes, por isso, não apresenta qualquer elemento de simulação. Esta interface está preparada para funcionar em ecrãs *touch* sem capacidade de *multi-touch* como o utilizado na HMI da célula.

Todas as variáveis trocadas com o *softPLC* são trocadas usando o protocolo ModBus.

Pela capacidade limitada deste protocolo de comunicação (relativamente à troca de texto) e pelos prazos apertados do projeto, que obrigaram à atribuição de prioridades às várias tarefas a fazer, esta interface tem um carácter mais simplista e funcional que a apresentada anteriormente.

7 Conclusões e Trabalhos Futuros

Esta dissertação teve como principais objetivos a orquestração do sistema de segurança da célula robótica e o projeto de uma solução de automação para o equipamento de movimentação e rotação de vigas.

Iniciou-se pelo estudo da estratégia de segurança a adotar, pela análise dos equipamentos da célula, pela revisão das normas a aplicar e, foi criada documentação de fácil acesso, com o intuito de servir como guia para futuros projetos. Essencialmente, esta documentação auxilia e acompanha os exercícios de avaliação e redução de riscos que constituem a base para a conceção de uma célula robótica segura para o Homem.

Adicionalmente, foi estudada e fundamentada uma solução de um problema detetado ao nível do processo produtivo, passando pela implementação de uma tarefa colaborativa entre o operador e o robô da célula.

Quanto aos Viradores, responsáveis pela movimentação e rotação de vigas, estes constituíram um desafio multidisciplinar, desde o projeto da sensorização e análise da estática do equipamento à programação e teste do mesmo.

Após escolhido o posicionamento dos sensores nos Viradores, foram conduzidos testes funcionais onde se verificou a validade das relações obtidas entre a medição dos sensores e a posição das diversas partes do equipamento. Assim, verificou-se que a análise efetuada descreve com sucesso o funcionamento real do equipamento.

Recolhidas todas as informações funcionais dos Viradores, foi realizada a sua programação, contemplando a diversidade de operações e especificidades deste equipamento. Orientada para a simplicidade, a programação do equipamento foi dividida em três partes correspondentes aos 3 estados do sistema: controlo manual, controlo automático e estado de emergência. Este último é compatibilizado com a programação do PLC de segurança realizada no âmbito do sistema de segurança da célula.

Os viradores foram testados e a interface de controlo foi integrada na aplicação de software da SARKKIS Robotics.

Assim, esta dissertação tem como produtos finais:

- Documentação de auxílio à conceção da estratégia de segurança de uma célula robótica.
- Um sistema de segurança, compatível com as normas de segurança industriais, capaz de controlar o acesso à célula, garantindo a segurança do Homem durante a fase de funcionamento da máquina.
- O controlo, manual e automático, dos Viradores de vigas, incluindo uma interface gráfica de controlo manual totalmente compatível com a HMI e com a aplicação de software da SARKKIS Robotics.

7.1 Trabalhos Futuros

No que respeita à componente relacionada com a segurança da célula, destacam-se algumas tarefas cuja concretização completará o trabalho até aqui realizado.

- Verificar, através de experimentação e tal como foi mencionado no subcapítulo 4.1.5, se o tempo de paragem dos equipamentos considerado na especificação das distâncias de segurança corresponde à realidade.
- Implementar a solução da tarefa colaborativa recorrendo ao Hand Guiding do robô estudada no subcapítulo 4.3.

No contexto da automação dos Viradores, destacam-se como sendo uma mais valia para o sistema, os seguintes pontos:

- Aumentar a performance dos viradores através da otimização dos movimentos, como por exemplo a utilização de ângulos mínimos.
- Acompanhar a evolução da célula de modo a detetar os seus pontos fracos, não esquecendo a opinião dos operadores e, se necessário, realizar modificações.
- Melhorar a interface gráfica no sentido de aumentar a informação disponível para o utilizador, à semelhança da interface gráfica do CodeSys, capítulo 6.
- Implementar, ao nível da programação, rotinas de calibração dos sensores uma vez que estes as permitem a realizar remotamente.

8 Referências

- [1] J. V. Escalé, “Human-robot interaction in the industry”, Master Thesis, Aalto University, 2015.
- [2] J. N. Pires, A. Loureiro, and G. Bolmsjo, *Welding Robots - Technology, System Issues and Applications*. London: Springer International Publishing.
- [3] “A Review on Recent Trends in Robotic Welding” *Certif. Int. J. Eng. Sci. Innov. Technol.*, vol. 9001, no. 5, pp. 2319–5967, 2008.
- [4] ISO. International Organization for Standarization, “ISO 12100-1” *Shock*, vol. 2003. pp. 1–8, 2002.
- [5] Rockwell Automation, “Machine Safety Guide”, Technical Report, Rockwell Automation, 2007.
- [6] ISO. International Organization for Standarization, “ISO 14121-1” vol. 1. p. 28, 2007.
- [7] ISO. International Organization for Standarization, “ISO 10218-2” vol. 2011. pp. 1–8, 2002.
- [8] ISO. International Organization for Standarization, “ISO/TR 14121-2” vol. 2012. 1994.
- [9] S. Robinson and M. Minstmc, “SIL or PL? What is the difference?”, Technical Report, Tuv-Sud.
- [10] IEC. International Electrotechnical Commission, “CEI IEC 62061”, 2005.
- [11] “How to do a 13849-1 analysis Archives - Machinery Safety 101”, Machinery Safety 101, [Online]. Available: <http://machinerysafety101.com/series/how-to-do-a-13849-1-analysis/>. [Accessed: 20-Feb-2017].
- [12] M. Ritchie and J. Monson, “Risk Assessment example”, Technical Report, Werner Electric, 2014.
- [13] ISO. International Organization for Standarization, “ISO 13855,” *Shock*, vol. 2010. pp. 1–8, 2002.
- [14] ISO. International Organization for Standarization, “ISO 13857:2008,” vol. 2008. p. 24, 2008.
- [15] “Human-robot collaboration | KUKA AG.”, KUKA, [Online]. Available: <https://www.kuka.com/en-us/technologies/mensch-roboter-kollaboration>. [Accessed: 28-Dec-2016].
- [16] “Human-robot collaboration | SICK”, SICK, [Online]. Available: <https://www.sick.com/se/en/-/human-robot-collaboration/w/human-robot-collaboration/>. [Accessed: 28-Dec-2016].
- [17] ISO. International Organization for Standarization, “ISO 10218-1”, 2006.
- [18] B. Matthias, “ISO / TS 15066 - Collaborative Robots Present Status”, Technical Report, ABB, 2015.
- [19] R. N. Shea, “ISO / TS 15066 Introduction What is ISO / TS 15066 ?”, Technical Report,

- TUV Rheinland, 2015.
- [20] “Fast Rotator”, Fast Rotator, [Online]. Available: <http://fastrotator.com/>. [Accessed: 19-Jul-2017].
 - [21] “Chain Rotator”, Stierli Bieger, [Online]. Available: <https://www.stierli-bieger.com/en/machinetype/9-rotator-turning-device.html>. [Accessed: 19-Apr-2017].
 - [22] “Ocean Flipper”, Ocean Machinery, [Online]. Available: <http://www.oceanmachinery.com/solutions/flipper-beam-rotator-column-positioner.html>. [Accessed: 19-Jul-2017].
 - [23] D. H. Hanssen, *Programmable logic controllers : a practical approach to IEC 61131-3 using CODESYS*. Wiley, 2015.
 - [24] “Definition of a PLC”, PLCdev, [Online]. Available: http://www.plcdev.com/definition_of_a_plc. [Accessed: 24-Apr-2017].
 - [25] W. Zhou, “Soft PLC Research And Development System Based On PC,” in *Proceedings of the 2015 International Conference on Intelligent Systems Research and Mechatronics Engineering*, 2015.
 - [26] M. Chmiel, E. Hryniewicz, A. Milik, and K. Zak, “Software Industrial Controller - soft PLC,” *IFAC Proc. Vol.* 36, no. 1, pp. 7–12, Feb. 2003.
 - [27] G. Yang and J. Yang, “Executing Strategy and Visualization Design for Instruction of Soft PLC System,” *J. Residuals Sci. Technol.*, vol. 14, no. 3, pp. 317–326, Apr. 2017.
 - [28] “Inside Machines: PC versus PLC: Comparing control options | Control Engineering”, Control Engineering [Online]. Available: <http://www.controleng.com/single-article/inside-machines-pc-versus-plc-comparing-control-options/9bf8690c6f23b11370bec90b52cb15c9.html>. [Accessed: 21-Apr-2017].
 - [29] R. Lewis, *Programming industrial control systems using IEC 1131-3*. IEE Control Engineering Series, 1998.
 - [30] K. H. John and M. Tiegelkamp, *IEC 61131-3: Programming Industrial Automation Systems*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010.
 - [31] PLCopen, “PLCopen for efficiency in automation”, [Online]. Available: http://www.plcopen.org/pages/promotion/publications/downloads/intro_iec_march2013.pdf/. [Accessed: 01-Sep-2017].
 - [32] “Coder’s Corner: The IEC 61131-3 Software Model”, Automation.com, [Online]. Available: <https://www.automation.com/library/articles-white-papers/coder146s-corner-the-iec-61131-3-software-model>. [Accessed: 22-Apr-2017].
 - [33] C. Moser, “Christian Moser’s WPF Tutorial”, WPFTutorial, [Online]. Available: <https://www.wpf-tutorial.net/Home.html>. [Accessed: 01-Sep-2017].
 - [34] “WPF vs. WinForms - The complete WPF tutorial”, WPF-Tutorial [Online]. Available: <http://www.wpf-tutorial.com/about-wpf/wpf-vs-winforms/>. [Accessed: 01-Sep-2017].
 - [35] “MA1440 - Arc Welding.” [Online]. Available: https://cdn2.hubspot.net/hubfs/366775/Blog_Robots/MA1440.pdf?t=1482272260730. [Accessed: 23-Dec-2016].
 - [36] “DX200 Robot Controller”, YASKAWA, [Online]. Available: https://cdn2.hubspot.net/hubfs/366775/PDFs/DX200_Controller.pdf?t=1482272260730. [Accessed: 23-Dec-2016].
 - [37] “Eixo Linear TSL YASKAWA MOTOMAN”, YASKAWA, [Online]. Available:

- https://www.roboplan.pt/upload/tsl-1036en_56bdd1d874c3e.pdf. [Accessed: 23-Dec-2016].
- [38] “Fronius TRANSSTEEL 3500/5000”, Fronius, [Online]. Available: http://www.fronius.com/cps/rde/xbcr/SID-40DDE107-E66B8C70/fronius_espana/M_06_0114_EN_Leaflet_TransSteel_3500_5000_125948_snapshot.pdf. [Accessed: 23-Dec-2016].
- [39] Motoman, “Motoman NX100 Controller ComArc Instruction Manual”, Yaskawa, 2007.
- [40] “Lasershop.de - ILP 622 LAN Laser Projector with network control” [Online]. Available: <http://www.lasershop.de/Laser-deflection-systems/laser-projectors/laser-projector-ilp622-572.html>. [Accessed: 01-Sep-2017].
- [41] Y. Motomedia, “MOTOMAN-MA1440 INSTRUCTIONS”, Technical Manual, YASKAWA, 2013.
- [42] “ReerR - Safety & Automation”, Reer, [Online]. Available: <http://www.reer.it/reer/en/products>. [Accessed: 04-May-2017].
- [43] “Carro de montagem - M1024006 KAISER+KRAFT Portugal”, Kaiser+Kraft, [Online]. Available: <https://www.kaiserkraft.pt/oficina-industria/carros-de-montagem/carro-de-montagem/p/M1024006/>. [Accessed: 27-Aug-2017].
- [44] “Catálogo - Blicle”, Blicle, [Online]. Available: http://catalogue.blickle.pt/#page_1. [Accessed: 27-Apr-2017].
- [45] “Rittal - The System”, Rittal, [Online]. Available: <https://www.rittal.com/pt-pt/content/pt/start/>. [Accessed: 27-Apr-2017].
- [46] “CODESYS – industrial IEC 61131-3 PLC programming”, 3S-Smart Software Solutions GmbH, [Online]. Available: <https://www.codesys.com/>. [Accessed: 02-Sep-2017].
- [47] “WAGO online catalog 750-352”, WAGO, [Online]. Available: https://eshop.wago.com/JPBC/0_5StartPage.jsp?supplierAID=750-352&catalogID=WAGO01&zone=7. [Accessed: 30-Apr-2017].
- [48] “SICK OD Value - DATASHEET”, SICK, [Online]. Available: https://www.sick.com/media/docs/3/53/353/quickstart_SENSICK_OD_Value_de_en_IM0031353.PDF. [Accessed: 30-Apr-2017].
- [49] “SICK DT20 HI - DATASHEET”, SICK, [Online]. Available: https://www.sick.com/media/dox/7/47/647/Operating_instructions_SENSICK_DT20_HI_de_en_fr_pt_da_it_nl_es_zh_IM0026647.PDF. [Accessed: 30-Apr-2017].
- [50] “WAGO online catalog 750-453”, WAGO, [Online]. Available: https://eshop.wago.com/JPBC/0_5StartPage.jsp?supplierAID=750-453&catalogID=WAGO01&zone=7. [Accessed: 02-Aug-2017].

ANEXO A: *Template* - Avaliação e Redução de riscos

Nome do projeto		
Versão	Data	Descrição

Célula robótica

Descrição da célula

.

Componentes

Componente	Observações

Layout inicial da célula

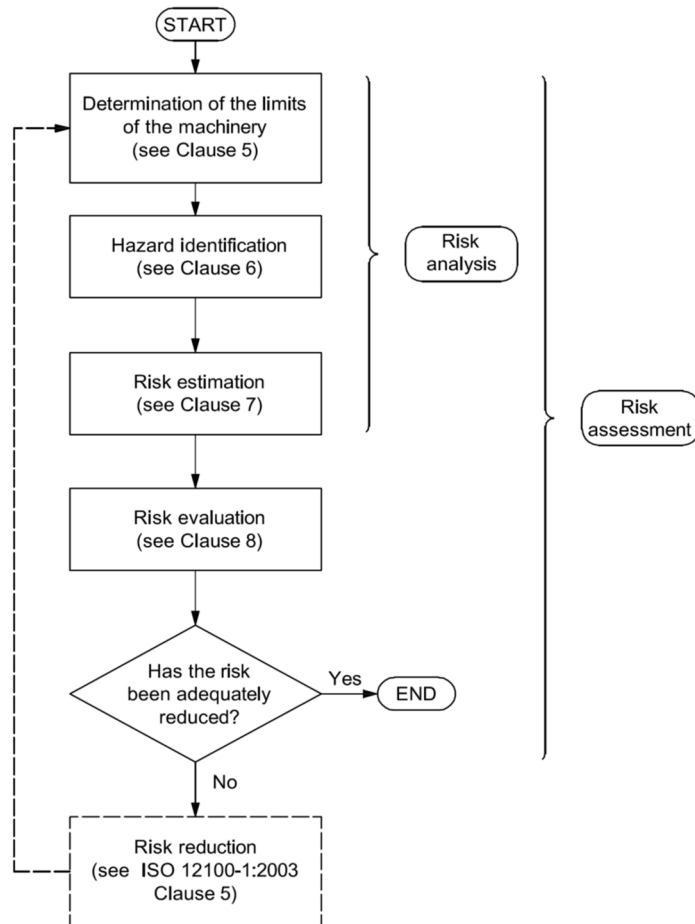
Experiências de uso anteriores

Risk Assessment

Metodologias utilizadas

A metodologia utilizada neste processo de avaliação dos riscos é descrita pela norma ISO 14121-1. A figura que se segue identifica as várias etapas deste processo.

Como método para avaliação do perigo é utilizada a metodologia proposta pela norma EN ISO 13849-1, classificando o perigo de acordo com o nível de performance requerido (PLr - Performance Level required).



Determinação de limites

Etapas e procedimento do processo operacional:

Limites de usabilidade

- Utilização prevista
- Má utilização razoavelmente previsível

Limites espaciais

Limites temporais

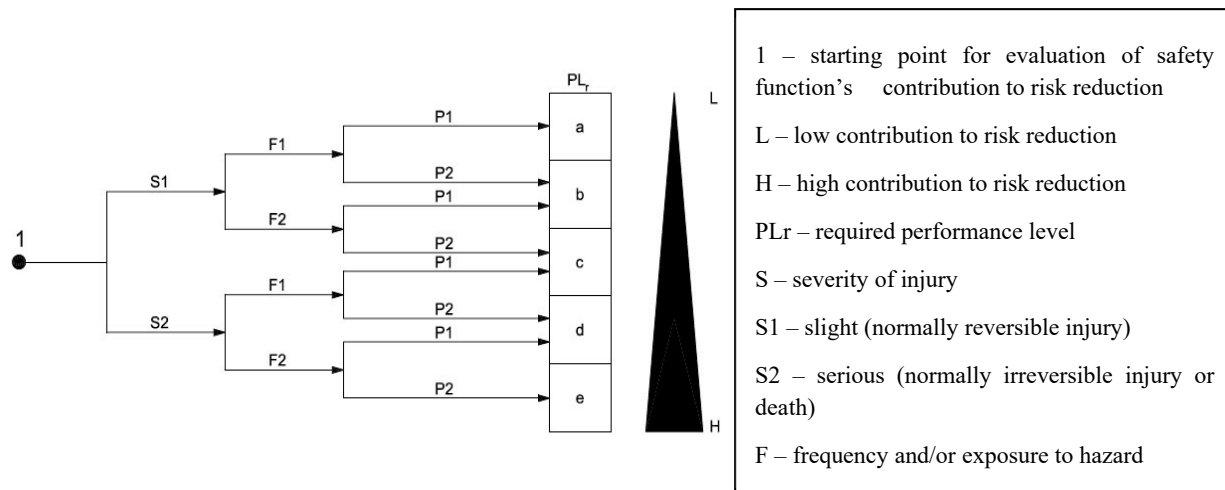
Outros limites

Identificação do perigo

Risk Assessment – Identificação do perigo					
Máquina (identificação)				Analista	
Referências				Versão	
Enquadramento (fase do ciclo de vida; parte/função da máquina)				Data	
Metodologia			ISO 10218-2:2011, Anexo A ISO 14121-1:2007, Anexo A ISO 12100		
Nº ref	Zona de perigo	Tarefa/Operação (ISO 14121-1:2007, Tabela A.3)	Perigo (ISO 14121-1:2007, Tabela A.1)	Situação perigosa (ISO 14121-1:2007, Tabela A.3)	Evento perigoso (ISO 14121-1:2007, Tabela A.4)
1					
2					
3					
4					
5					
6					
7					
8					
9					

Estimativa, avaliação e redução do risco

Metodologia segundo a norma EN ISO 13849-1, classificando o perigo de acordo com o nível de performance requerido (PLr - Performance Level required). A figura seguinte ilustra em maior detalhe este método.



Nº ref	Estimativa do risco (inicial)				Redução do risco	Estimativa do risco residual (depois da redução do risco)				Observações
	S	F	P	PLr		S	F	P	PLr	
1										
2										
3										
4										
5										
6										
7										
8										
9										

ANEXO B: Projeto dos suportes para os lasers

Em conjunto com a equipa da SARKKIS Robotics decidiu-se que estes suportes seriam fabricados recorrendo à impressão 3D.

Os suportes foram desenhados com os seguintes requisitos:

- Conferirem espaçamento suficiente
- Serem fáceis de instalar em ambos os flippers
- Permitirem o ajusto do laser

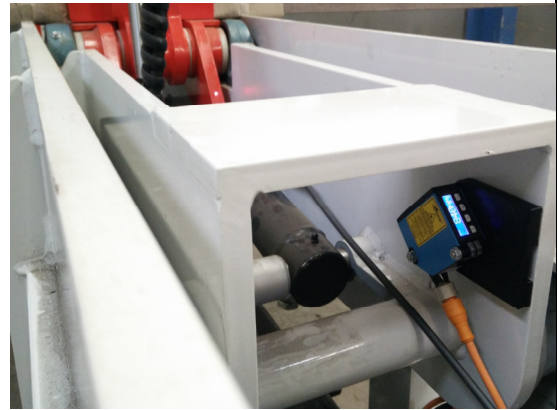
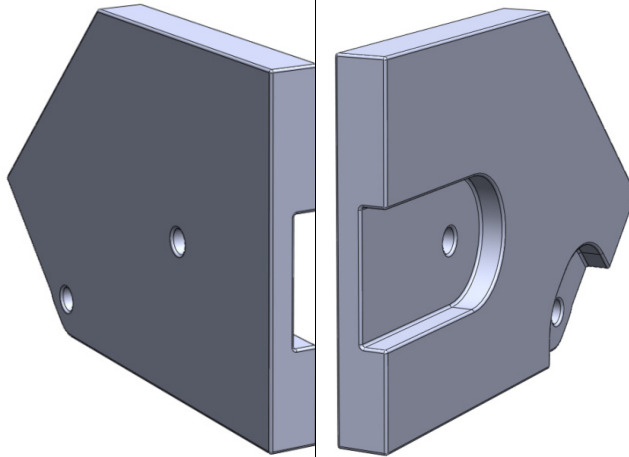
A Tabela 30 apresenta os suportes projetados para cada um dos lasers. As duas colunas da esquerda são representações CAD onde se vê a face que fica em contacto, na primeira coluna, e a face que fica em contacto com a parede, na segunda coluna.

Na face que fica em contacto com a parede foi tido o cuidado de recortar saliências para permitir a entrada de uma chave de bocas para ser possível montar e desmontar os sensores ao suporte.

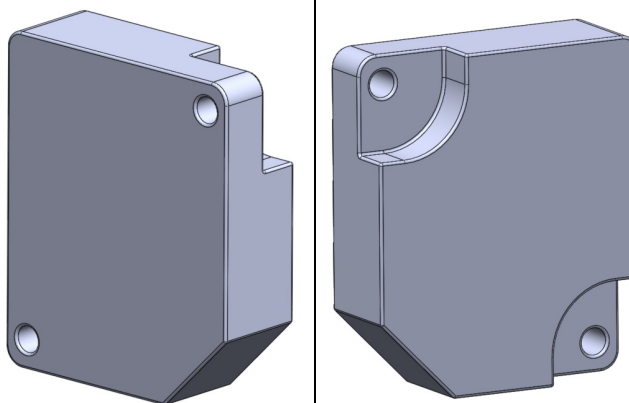
A coluna da direita apresenta uma fotografia do respetivo sensor instalado no Virador.

Tabela 30 - Projeto dos suportes dos Lasers

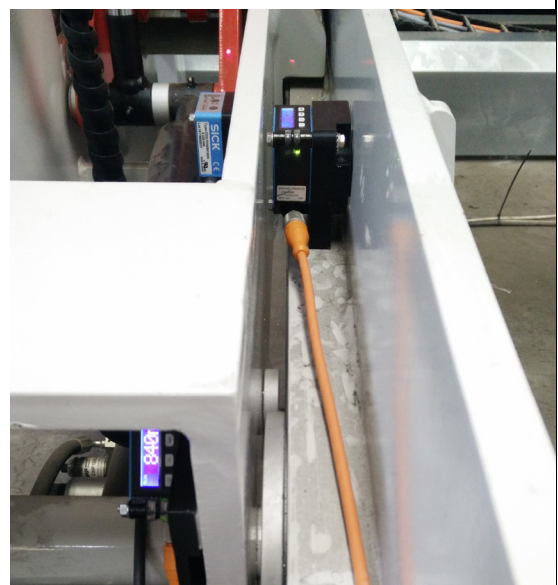
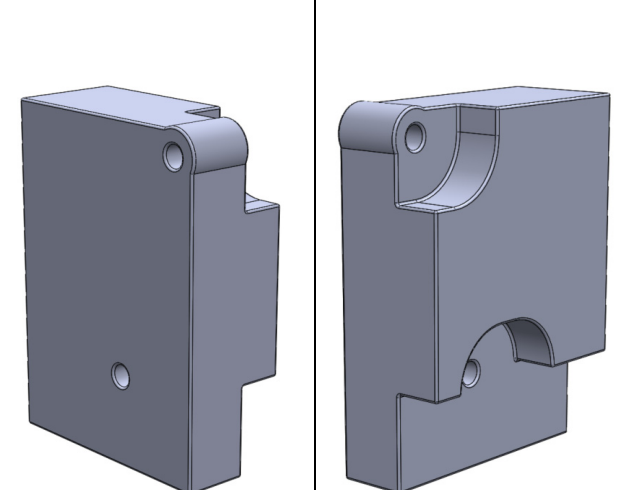
Laser 1



Laser 2



Laser 3



ANEXO C: Programa de controlo dos Viradores

Programa Main

Neste programa está incluída a programação do modo manual e do estado de emergência.

Declaração de variáveis

```
// PLC de Segunrança
Emergencia_SPLC : BOOL;
Enable_SPLC : BOOL;
Seletor_SPLC : BOOL;

RotateDim_R_Trig : R_Trig;
RotateDim : INT;
BeamHorDim : REAL;
BeamVerDim : REAL;

Selector : BOOL ;
AUTO_LED : BOOL;
MANUAL_LED : BOOL;
EMERGENCIA_LED : BOOL;

StateDesc1 : STRING;
StateDesc2 : STRING;

// valores das cartas WAGO - Lasers
    fl_right_highbyte : BYTE;
    fl_right_lowbyte : BYTE;
    fl_left_highbyte : BYTE;
    fl_left_lowbyte : BYTE;
    fl_horizontal_highbyte : BYTE;
    fl_horizontal_lowbyte : BYTE;
```

```
f2_right_highbyte : BYTE;  
f2_right_lowbyte : BYTE;  
f2_left_highbyte : BYTE;  
f2_left_lowbyte : BYTE;  
f2_horizontal_highbyte : BYTE;  
f2_horizontal_lowbyte : BYTE;
```

```
f1_left_value : REAL;  
f1_right_value : REAL;  
f1_horizontal_value : REAL;
```

```
f2_left_value : REAL;  
f2_right_value : REAL;  
f2_horizontal_value : REAL;
```

```
f1_left_degree : REAL;  
f1_right_degree : REAL;  
f1_horizontal_mm : REAL;
```

```
f2_left_degree : REAL;  
f2_right_degree : REAL;  
f2_horizontal_mm : REAL;
```

```
// Valores Max e Min - Calibração
```

```
Erro : REAL;
```

```
f1_left_MaxValue : REAL;  
f1_left_MedValue : REAL;  
f1_left_MinValue : REAL;
```

```
f1_right_MaxValue : REAL;  
f1_right_MedValue : REAL;  
f1_right_MinValue : REAL;
```

```
f1_horizontal_MaxValue : REAL;  
f1_horizontal_MinValue : REAL;
```



```

f2_left_MaxValue : REAL;
f2_left_MedValue : REAL;
f2_left_MinValue : REAL;

f2_right_MaxValue : REAL;
f2_right_MedValue : REAL;
f2_right_MinValue : REAL;

f2_horizontal_MaxValue : REAL;
f2_horizontal_MinValue : REAL;

left_slider : REAL;
right_slider : REAL;
horizontal_slider : REAL;

// Instanciação do controlo por solenoide
fbFlipper1_Sol : FB_FLIPPER_SOL;
fbFlipper2_Sol : FB_FLIPPER_SOL;

// Instanciação do controlo por ações - Macros
fbFlipper1_Act : FB_FLIPPER_ACT;
fbFlipper2_Act : FB_FLIPPER_ACT;

// Instanciação do funcionamento em automático
fbAuto : FB_AUTOMATIC;

// Modos de funcionamento
MODE : INT := 0;
EMERGENCIA : BOOL;
Rearme : BOOL;
MANUAL : BOOL;
AUTO : BOOL;
ModeDesc : STRING := '';

// ModBus Variables - Inputs
EMERGENCIA_MB : BOOL;

```

StateF1_MB : BOOL; // coil 10 - Input 0 bit 10 - Flipper 1 bomba

StateF2_MB : BOOL; // coil 20 - Input 1 bit 04 - Flipper 2 bomba

Af1_MB : BOOL; // coil 11 - Input 0 bit 11 - Flipper 1 sol. A

Cf1_MB : BOOL; // coil 12 - Input 0 bit 12 - Flipper 1 sol. C

Bf1_MB : BOOL; // coil 13 - Input 0 bit 13 - Flipper 1 sol. B

Gf1_MB : BOOL; // coil 14 - Input 0 bit 14 - Flipper 1 sol. G

Ff1_MB : BOOL; // coil 15 - Input 0 bit 15 - Flipper 1 sol. F

Kf1_MB : BOOL; // coil 16 - Input 1 bit 00 - Flipper 1 sol. K

Jf1_MB : BOOL; // coil 17 - Input 1 bit 01 - Flipper 1 sol. J

Af2_MB : BOOL; // coil 21 - Input 1 bit 05 - Flipper 2 sol. A

Cf2_MB : BOOL; // coil 22 - Input 1 bit 06 - Flipper 2 sol. C

Bf2_MB : BOOL; // coil 23 - Input 1 bit 07 - Flipper 2 sol. B

Gf2_MB : BOOL; // coil 24 - Input 1 bit 08 - Flipper 2 sol. G

Ff2_MB : BOOL; // coil 25 - Input 1 bit 09 - Flipper 2 sol. F

Kf2_MB : BOOL; // coil 26 - Input 1 bit 10 - Flipper 2 sol. K

Jf2_MB : BOOL; // coil 27 - Input 1 bit 11 - Flipper 2 sol. J

SuspF1_MB : BOOL;

SuspF2_MB : BOOL;

PouF1_MB : BOOL;

PouF2_MB : BOOL;

RotPF1_MB : BOOL;

RotPF2_MB : BOOL;

RotNF1_MB : BOOL;

RotNF2_MB : BOOL;

BeamHorDim_MB : WORD := 125; // Dimensão horizontal da viga

BeamVerDim_MB : WORD := 300; // Dimensão vertical da iga

Encostar_MB : BOOL;

Rodar_MB : BOOL;

Home_MB : BOOL;

// HMI Variables - Inputs

```

StateF1_HMI : BOOL; // coil 10 - Input 0 bit 10 - Flipper 1 bomba
StateF2_HMI : BOOL; // coil 20 - Input 1 bit 04 - Flipper 2 bomba

```

```

A_HMI : BOOL; // coil 11 - Input 0 bit 11 - Flipper 1 sol. A
C_HMI : BOOL; // coil 12 - Input 0 bit 12 - Flipper 1 sol. C
B_HMI : BOOL; // coil 13 - Input 0 bit 13 - Flipper 1 sol. B
G_HMI : BOOL; // coil 14 - Input 0 bit 14 - Flipper 1 sol. G
F_HMI : BOOL; // coil 15 - Input 0 bit 15 - Flipper 1 sol. F
K_HMI : BOOL; // coil 16 - Input 1 bit 00 - Flipper 1 sol. K
J_HMI : BOOL; // coil 17 - Input 1 bit 01 - Flipper 1 sol. J

```

```

Susp_HMI : BOOL;
Pou_HMI : BOOL;
RotP_HMI : BOOL;
RotN_HMI : BOOL;

```

```

// BeamHorDim_MB : REAL; // Dimensão horizontal da viga
Encostar_HMI : BOOL;
Rodar_HMI : BOOL;
Home_HMI : BOOL;

BF_HMI : BOOL;
CG_HMI : BOOL;

```

Programa

```

f1_right_value := WORD_TO_REAL(SHL(BYTE_TO_WORD(f1_right_highbyte),8) +
BYTE_TO_WORD(f1_right_lowbyte));
f1_left_value  := WORD_TO_REAL(SHL(BYTE_TO_WORD(f1_left_highbyte),8) +
BYTE_TO_WORD(f1_left_lowbyte));
f1_horizontal_value                                     :=
WORD_TO_REAL(SHL(BYTE_TO_WORD(f1_horizontal_highbyte),8) +
BYTE_TO_WORD(f1_horizontal_lowbyte));

f2_right_value := WORD_TO_REAL(SHL(BYTE_TO_WORD(f2_right_highbyte),8) +
BYTE_TO_WORD(f2_right_lowbyte));

```

```

f2_left_value := WORD_TO_REAL(SHL(BYTE_TO_WORD(f2_left_highbyte),8) +
BYTE_TO_WORD(f2_left_lowbyte));

f2_horizontal_value :=
WORD_TO_REAL(SHL(BYTE_TO_WORD(f2_horizontal_highbyte),8) +
BYTE_TO_WORD(f2_horizontal_lowbyte));

(*)
// Sliders para testes
f1_left_value := f2_left_value := 32764*left_slider;
f1_right_value := f2_right_value := 32764*right_slider;
f1_horizontal_value := f2_horizontal_value := 32764*horizontal_slider;
*)

f1_left_degree := LASERVALUE_TO_DEGREE_LEFT(f1_left_value);
f1_right_degree := LASERVALUE_TO_DEGREE_RIGHT(f1_right_value);
f1_horizontal_mm := LASERVALUE_TO_MM_HORIZONTAL(f1_horizontal_value);

f2_left_degree := LASERVALUE_TO_DEGREE_LEFT(f2_left_value);
f2_right_degree := LASERVALUE_TO_DEGREE_RIGHT(f2_right_value);
f2_horizontal_mm := LASERVALUE_TO_MM_HORIZONTAL(f2_horizontal_value);

// Todos os solenoides a falso no inicio
fbFlipper1_Sol(FlipperID:=1 , State:=FALSE, C_In:=FALSE , B_In:=FALSE , G_In:=FALSE
, F_In:=FALSE , K_In:=FALSE , J_In:=FALSE);
fbFlipper2_Sol(FlipperID:=2 , State:=FALSE, C_In:=FALSE , B_In:=FALSE , G_In:=FALSE
, F_In:=FALSE , K_In:=FALSE , J_In:=FALSE);

fbFlipper1_Act ( bSusp := FALSE , bPou := FALSE , bRotP := FALSE , bRotN := FALSE ,
bPou_Short := FALSE , bHor := FALSE);
fbFlipper1_Act ( bSusp := FALSE , bPou := FALSE , bRotP := FALSE , bRotN := FALSE ,
bPou_Short := FALSE , bHor := FALSE);

// Atribuições modo Manual
// Atribuições dos valores máximos e mínimos a cada flipper
fbFlipper1_Act ( Left_MaxValue := f1_left_MaxValue , Left_MedValue :=
f1_left_MedValue , Left_MinValue := f1_left_MinValue );
fbFlipper1_Act ( Right_MaxValue := f1_right_MaxValue , Right_MedValue :=
f1_right_MedValue , Right_MinValue := f1_right_MinValue );
fbFlipper1_Act ( Erro := Erro );

```

```

        fbFlipper2_Act ( Left_MaxValue := f2_left_MaxValue , Left_MedValue :=
f2_left_MedValue , Left_MinValue := f2_left_MinValue );

        fbFlipper2_Act ( Right_MaxValue := f2_right_MaxValue , Right_MedValue :=
f2_right_MedValue , Right_MinValue := f2_right_MinValue );

        fbFlipper2_Act ( Erro := Erro );

        // Atribuição dos valores medidos pelos laser

        fbFlipper1_Act ( Left_Value := f1_left_value , Right_Value := f1_right_value ,
Horizontal_Value_mm := f1_horizontal_mm);

        fbFlipper2_Act ( Left_Value := f2_left_value , Right_Value := f2_right_value ,
Horizontal_Value_mm := f2_horizontal_mm);

// Atribuições modo Auto

        // Atribuições dos valores máximos e mínimos a cada flipper

        fbAuto.fbFlipper1_Act ( Left_MaxValue := f1_left_MaxValue , Left_MedValue :=
f1_left_MedValue , Left_MinValue := f1_left_MinValue );

        fbAuto.fbFlipper1_Act ( Right_MaxValue := f1_right_MaxValue , Right_MedValue :=
f1_right_MedValue , Right_MinValue := f1_right_MinValue );

        fbAuto.fbFlipper1_Act ( Erro := Erro );

        fbAuto.fbFlipper2_Act ( Left_MaxValue := f2_left_MaxValue , Left_MedValue :=
f2_left_MedValue , Left_MinValue := f2_left_MinValue );

        fbAuto.fbFlipper2_Act ( Right_MaxValue := f2_right_MaxValue , Right_MedValue :=
f2_right_MedValue , Right_MinValue := f2_right_MinValue );

        fbAuto.fbFlipper2_Act ( Erro := Erro );

        // Atribuição dos valores medidos pelos laser

        fbAuto.fbFlipper1_Act ( Left_Value := f1_left_value , Right_Value := f1_right_value ,
Horizontal_Value_mm := f1_horizontal_mm);

        fbAuto.fbFlipper2_Act ( Left_Value := f2_left_value , Right_Value := f2_right_value ,
Horizontal_Value_mm := f2_horizontal_mm);

// Modos de funcionamento
IF NOT Seletor_SPLC THEN
    MANUAL := TRUE;
    AUTO := FALSE;
ELSE
    MANUAL := FALSE;
    AUTO := TRUE;
END_IF

CASE MODE OF
0:

```

```

IF EMERGENCIA_SPLC OR EMERGENCIA OR EMERGENCIA_MB THEN
    MODE := 30;
ELSIF AUTO THEN
    MODE := 20;
ELSIF MANUAL THEN
    MODE := 10;
END_IF

```

10: // MANUAL

```

ModeDesc := 'MANUAL';
IF NOT MANUAL THEN
    MODE := 0;
END_IF
IF EMERGENCIA_SPLC THEN
    MODE := 30;
END_IF
AUTO_LED := FALSE;
MANUAL_LED := TRUE;
EMERGENCIA_LED := FALSE;

```

```

fbAuto (bReset := TRUE);

```

```

fbFlipper1_Sol(State:=StateF1_MB OR StateF1_HMI);
fbFlipper2_Sol(State:=StateF2_MB OR StateF2_HMI);

```

// Ativação de solenoides

// if serve para não ser possível ativar uma coil enquanto se está dentro de uma ação

```

IF NOT fbFlipper1_Act.State THEN
    fbFlipper1_Sol(C_In := Cf1_MB OR (C_HMI AND StateF1_HMI) OR
(CG_HMI AND StateF1_HMI) , B_In := Bf1_MB OR (B_HMI AND StateF1_HMI) OR
(BF_HMI AND StateF1_HMI) , G_In := Gf1_MB OR (G_HMI AND StateF1_HMI) OR
(CG_HMI AND StateF1_HMI) , F_In := Ff1_MB OR (F_HMI AND StateF1_HMI) OR
(BF_HMI AND StateF1_HMI) , K_In := Kf1_MB OR (K_HMI AND StateF1_HMI) , J_In :=
Jf1_MB OR (J_HMI AND StateF1_HMI) );

```

```

END_IF

```

```

IF NOT fbFlipper2_Act.State THEN

```

```

    fbFlipper2_Sol(C_In := Cf2_MB OR (C_HMI AND StateF2_HMI) OR
(CG_HMI AND StateF2_HMI) , B_In := Bf2_MB OR (B_HMI AND StateF2_HMI) OR
(BF_HMI AND StateF2_HMI) , G_In := Gf2_MB OR (G_HMI AND StateF2_HMI) OR

```

```
(CG_HMI AND StateF2_HMI) , F_In := Ff2_MB OR (F_HMI AND StateF2_HMI) OR
(BF_HMI AND StateF2_HMI) , K_In := Kf2_MB OR (K_HMI AND StateF2_HMI) , J_In :=
Jf2_MB OR (J_HMI AND StateF2_HMI);
```

```
END_IF
```

```
// Inicialização de ações
```

```
fbFlipper1_Act ( bReset := NOT fbFlipper1_Sol.State , bSusp := SuspF1_MB OR
(Susp_HMI AND StateF1_HMI) , bPou := PouF1_MB OR (Pou_HMI AND StateF1_HMI) ,
bRotP := RotPF1_MB OR (RotP_HMI AND StateF1_HMI) , bRotN := RotNF1_MB OR
(RotN_HMI AND StateF1_HMI) );
```

```
fbFlipper2_Act ( bReset := NOT fbFlipper2_Sol.State , bSusp := SuspF2_MB OR
(Susp_HMI AND StateF2_HMI) , bPou := PouF2_MB OR (Pou_HMI AND StateF2_HMI) ,
bRotP := RotPF2_MB OR (RotP_HMI AND StateF2_HMI) , bRotN := RotNF2_MB OR
(RotN_HMI AND StateF2_HMI) );
```

```
StateDesc1 := fbFlipper1_Act.StateDesc;
```

```
StateDesc2 := fbFlipper2_Act.StateDesc;
```

```
// Não é necessário o If, mas fica em conformidade com o anterior
```

```
IF fbFlipper1_Act.State THEN
```

```
fbFlipper1_Sol(C_In := fbFlipper1_Act.C , B_In := fbFlipper1_Act.B , G_In :=
fbFlipper1_Act.G , F_In := fbFlipper1_Act.F , K_In := fbFlipper1_Act.K , J_In :=
fbFlipper1_Act.J );
```

```
END_IF
```

```
IF fbFlipper2_Act.State THEN
```

```
fbFlipper2_Sol(C_In := fbFlipper2_Act.C , B_In := fbFlipper2_Act.B , G_In :=
fbFlipper2_Act.G , F_In := fbFlipper2_Act.F , K_In := fbFlipper2_Act.K , J_In :=
fbFlipper2_Act.J );
```

```
END_IF
```

```
20: // AUTO
```

```
ModeDesc := 'AUTO';
```

```
IF NOT AUTO THEN
```

```
MODE := 0;
```

```
END_IF
```

```
IF EMERGENCIA_SPLC THEN
```

```
MODE := 30;
```

```
END_IF
```

```
AUTO_LED := TRUE;
```

```
MANUAL_LED := FALSE;
```

```
EMERGENCIA_LED := FALSE;
```

```

//Bombas dos flippers iniciam apenas quando é necessário realizar alguma operação

fbFlipper1_Sol(State:=fbAuto.State);
fbFlipper2_Sol(State:=fbAuto.State);

// Inputs

fbAuto (bReset := FALSE , bHome := Home_HMI OR Home_MB , bEncostar :=
Encostar_HMI OR Encostar_MB , bRodar := Rodar_HMI OR Rodar_MB , BeamHorDim :=
BeamHorDim , BeamVerDim := BeamVerDim );

StateDesc1 := fbAuto.fbFlipper1_Act.StateDesc;
StateDesc2 := fbAuto.fbFlipper2_Act.StateDesc;

fbFlipper1_Sol(C_In := fbAuto.fbFlipper1_Act.C , B_In := fbAuto.fbFlipper1_Act.B ,
G_In := fbAuto.fbFlipper1_Act.G , F_In := fbAuto.fbFlipper1_Act.F , K_In :=
fbAuto.fbFlipper1_Act.K , J_In := fbAuto.fbFlipper1_Act.J );
fbFlipper2_Sol(C_In := fbAuto.fbFlipper2_Act.C , B_In := fbAuto.fbFlipper2_Act.B ,
G_In := fbAuto.fbFlipper2_Act.G , F_In := fbAuto.fbFlipper2_Act.F , K_In :=
fbAuto.fbFlipper2_Act.K , J_In := fbAuto.fbFlipper2_Act.J );

30: // EMERGÊNCIA
ModeDesc := 'EMERGÊNCIA';
IF NOT EMERGENCIA_SPLC AND MANUAL THEN
    MODE := 10;
    EMERGENCIA := FALSE;
ELSIF NOT EMERGENCIA_SPLC AND AUTO THEN
    MODE := 20;
    EMERGENCIA := FALSE;
END_IF
AUTO_LED := FALSE;
MANUAL_LED := FALSE;
EMERGENCIA_LED := TRUE;

fbFlipper1_Act (bReset := TRUE);
fbFlipper2_Act (bReset := TRUE);
fbAuto (bReset := TRUE);
END_CASE

```


Movimentos elementares

Declaração de variáveis

VAR_INPUT

```
FlipperID : INT; // 1 or 2
State : BOOL;
C_In : BOOL; // Right UP
B_In : BOOL; // Right DOWN
G_In : BOOL; // Left UP
F_In : BOOL; // Left DOWN
K_In : BOOL; // Esquerda
J_In : BOOL; // Direita
```

END_VAR

VAR_OUTPUT

```
A_Out : BOOL; // Enable
C_Out : BOOL; // Right UP
B_Out : BOOL; // Right DOWN
G_Out : BOOL; // Left UP
F_Out : BOOL; // Left DOWN
K_Out : BOOL; // Esquerda
J_Out : BOOL; // Direita
```

END_VAR

Programa

IF State THEN

```
C_Out := C_In;
B_Out := B_In;
G_Out := G_In;
F_Out := F_In;
K_Out := K_In;
J_Out := J_In;
```

ELSE

```
C_Out := FALSE;
B_Out := FALSE;
G_Out := FALSE;
```

```
        F_Out := FALSE;
        K_Out := FALSE;
        J_Out := FALSE;
    END_IF

    IF C_Out OR B_Out OR G_Out OR F_Out OR K_Out OR J_Out THEN
        A_Out := TRUE;
    ELSE
        A_Out := FALSE;
    END_IF
```

Operações elementares

Declaração de variáveis

VAR_INPUT

// Inicialização de operações

bReset : BOOL;

bSusp : BOOL;

bsusp_Short : BOOL; // fica a pá esquerda a 90°

bPou : BOOL;

bPou_Short : BOOL; // versão short

bRotP : BOOL;

bRotP_Susp : BOOL; // Fazer rotação depois de suspender

bRotN : BOOL;

bHor : BOOL; //mover para a direita

bHome : BOOL;

// Variaveis

Left_Value : REAL;

Left_MaxValue : REAL;

Left_MedValue: REAL;

Left_MinValue : REAL;

Right_Value : REAL;

Right_MaxValue : REAL;

Right_MedValue : REAL;

Right_MinValue : REAL;

Horizontal_Value_mm : REAL;

Hor_PV : REAL;

Hor_Margin : REAL;

Erro : REAL;

END_VAR

VAR_OUTPUT

```

Done : BOOL; // sinaliza o final de uma macro
State : BOOL; // indica Virador livre (TRUE) ou em funcionamento (FALSE)
A : BOOL; // Enable
C : BOOL; // Right UP
B : BOOL; // Right DOWN
G : BOOL; // Left UP
F : BOOL; // Left DOWN
K : BOOL; // Esquerda
J : BOOL; // Direita

END_VAR

VAR
    LastOperation : INT; // 1-Mov. Horizontal , 2-Suspensde , 3-Pousar , 4-PousarShort , 5-
Rotação+ , 6-RotaçãoSusten (Não assinalado) , 7-Rotação- , 8-Home

// Start bit's de cada macro
    bHor_R_Trig : R_Trig;
    bSusp_R_Trig : R_Trig;
    bPou_R_Trig : R_Trig;
    bPou_Short_R_Trig : R_Trig;
    bRotP_R_Trig : R_Trig;
    bRotP_Susp_R_Trig : R_Trig;
    bRotN_R_Trig : R_Trig;
    bHome_R_Trig : R_Trig;

// Macros
    iHor : INT := 0;
    iSusp : INT := 0; // Case de posicionamento inicial grosseiro. Levanta e desloca para
perto do batente
    iPou : INT := 0; // Case de posicionamento inicial fino . Aproxima do batente
    iRotP : INT := 0; // Rodar 90° sentido positivo
    iRotN : INT := 0; // Rodar 90° sentido negativo
    iHome : INT := 0;

    StateDesc : STRING := '';

    Pou_Short : BOOL;

```

```

    Susp_Short : BOOL;
    RotP_Susp : BOOL;
END_VAR

```

Programa

```

bHor_R_Trig (CLK := bHor);
bSusp_R_Trig (CLK := bSusp);

```

```

bPou_R_Trig (CLK := bPou);
bPou_Short_R_Trig (CLK := bPou_Short);
bRotP_R_Trig (CLK := bRotP);
bRotP_Susp_R_Trig (CLK := bRotP_Susp);
bRotN_R_Trig (CLK := bRotN);
bHome_R_Trig (CLK := bHome);

```

```

IF bReset THEN
    iSusp := iPou := iRotP := iRotN := iHor := iHome := 0;
    StateDesc := '';
    State := FALSE;
    Pou_Short := FALSE;
    RotP_Susp := FALSE;
END_IF
IF iSusp = 0 AND iPou = 0 AND iRotP = 0 AND iRotN = 0 AND iHor = 0 AND iHome = 0
THEN
    State := FALSE;
    A := C := B := G := F := K := J := FALSE;
END_IF

```

```

CASE iHor OF
    0:
        IF bHor_R_Trig.Q AND NOT State THEN
            iHor := 10;
            State := TRUE;
        END_IF
    10:
        IF Horizontal_Value_mm < Hor_PV + Hor_Margin THEN
            iHor := 20; // tem de andar para a direita
        END_IF
        IF Horizontal_Value_mm > Hor_PV + Hor_Margin THEN
            iHor := 25; // tem de andar para a esquerda
        END_IF
    20:

```

```

        StateDesc := 'Deslocar para a Esquerda';
        J := TRUE;
        IF Horizontal_Value_mm > Hor_PV + Hor_Margin THEN
            J := FALSE;
            iHor := 0;
            LastOperation := 1;
            State := FALSE;
            StateDesc := '';
        END_IF
25:
        StateDesc := 'Deslocar para a Direita';
        K := TRUE;
        IF Horizontal_Value_mm < Hor_PV + Hor_Margin THEN
            K := FALSE;
            iHor := 0;
            State := FALSE;
            StateDesc := '';
            LastOperation := 1;
        END_IF
END_CASE

CASE iSusp OF // Suspende
0:
        IF bSusp_R_Trig.Q AND NOT State AND LastOperation <> 2 THEN
            iSusp := 10;
            State := TRUE;
        END_IF
10:
        // Left UP de 0° até 90° - talvez rodar só 45 e acrescentar caso 25
        // Considera-se estar Right_0 e Left_0
        StateDesc := 'Left UP de 0° até 90°';
        C := TRUE;
        IF Left_Value > Left_MaxValue - Erro THEN
            C := FALSE;
            iSusp := 20;
            StateDesc := '';
            // Timer(IN:=FALSE);
        END_IF
20:
        // Right UP de 0° até 45°
        // Considera-se estar Right_0 e Left_90
        StateDesc := 'Right UP de 0° até 45°';
        // Timer(IN:=TRUE, PT:=TLU45);
        G := TRUE;
        IF Right_Value > Right_MedValue - Erro THEN // falta margem para o outro
lado, ou não
            G := FALSE;
            iSusp := 0;
            State := FALSE;
            StateDesc := '';
            LastOperation := 2;
            // Timer(IN:=FALSE);

```

```

        END_IF
    END_CASE

CASE iPou OF // Pousar
0:
    IF (bPou_R_Trig.Q OR bPou_Short_R_Trig.Q) AND NOT State THEN
        iPou := 10;
        State := TRUE;
        IF bPou_Short_R_Trig.Q THEN //Versão curta - Sem pousar pá
esquerda no final
            Pou_Short := TRUE;
        END_IF
    END_IF

10:
    // Right Down de 45° até 0°
    // Considera-se estar Right_45 e Left_45(90 rel)
    StateDesc := 'Right Down de 45° até 0°';
    // Timer(IN:=TRUE, PT:=TLD45);
    F := TRUE;
    IF Right_Value < Right_MinValue + Erro - 100 THEN
        F := FALSE;
        IF Pou_Short THEN
            iPou := 0;
            State := FALSE;
            Pou_Short := FALSE;
            LastOperation := 4;
        ELSE
            iPou := 20;
        END_IF
        StateDesc := ' ';
        // Timer(IN:=FALSE);
    END_IF

20:
    // Left Down de 90° até 0°
    // Considera-se estar Right_0 e Left_90
    StateDesc := 'Left Down de 90° até 0°';
    // Timer(IN:=TRUE, PT:=TRD90);
    B := TRUE;
    IF Left_Value < Left_MinValue + Erro - 100 THEN
        B := FALSE;
        iPou := 0;
        State := FALSE;
        StateDesc := ' ';
        LastOperation := 3;
        // Timer(IN:=FALSE);
    END_IF
END_CASE

CASE iRotN OF // Rotação +90°
0:
    IF bRotN_R_Trig.Q AND NOT State THEN

```

```

        iRotN := 10;
        State := TRUE;
    END_IF
10:
// Right UP de 0° até 90°
// Para Right UP antes da Left é preciso fazer ao mesmo tempo Left DOWN
// Considera-se Right_0 e Left_0
// Considera-se eixo rotação à direita da viga
    StateDesc := 'Right UP de 0° até 90°';
//    Timer(IN:=TRUE, PT:=TLRU);
    G := TRUE;
    C := TRUE;
    IF Right_Value > Right_MaxValue - Erro THEN
        G := FALSE;
        C := FALSE;
        iRotN := 20;
        StateDesc := ' ';
//        Timer(IN:=FALSE);
    END_IF
20:
// Right DOWN de 90° até 0°
// Desta vez a pá da esquerda não será atuada, ela vai subir sozinha
// Considera-se estar Right_90 e Left_0(90 rel)
    StateDesc := 'Right DOWN de 90° até 0°';
//    Timer(IN:=TRUE, PT:=TLD90);
    F := TRUE;
    IF Right_Value < Right_MinValue + Erro - 100 THEN
        F := FALSE;
        iRotN := 30;
        StateDesc := ' ';
//        Timer(IN:=FALSE);
    END_IF
30:
// Left DOWN de 90° até 0°
// Esta etapa pode ser feita ao mesmo tempo da anterior
// Considera-se estar Right_0 e Left_90
    StateDesc := 'Left DOWN de 90° até 0°';
//    Timer(IN:=TRUE, PT:=TRD90);
    B := TRUE;
    IF Left_Value < Left_MinValue + Erro -100 THEN
        B := FALSE;
        iRotN := 0;
        State := FALSE;
        StateDesc := ' ';
        LastOperation := 7;
//        Timer(IN:=FALSE);
    END_IF
END_CASE

CASE iRotP OF // Rotação -90°
0:
    IF (bRotP_R_Trig.Q OR bRotP_Susp_R_Trig.Q) AND NOT State THEN

```



```

        iRotP := 10;
        State := TRUE;
        IF bRotP_Susp_R_Trig.Q THEN
            iRotP := 20;
        END_IF
    END_IF

10:
    // Left UP de 0° até 90°
    // Considera-se Right_0 e Left_0
    // Considera-se eixo rotação à esquerda da viga
    StateDesc := 'Left UP de 0° até 90°';
    C := TRUE;

    IF Left_Value > Left_MaxValue - Erro THEN
        C := FALSE;
        // F := FALSE;
        // RotP_Susp := FALSE;
        iRotP := 20;
        StateDesc := ' ';
    END_IF

20:
    // Right UP de 0° até 90°
    // Considera-se estar Right_0 e Left_90
    StateDesc := 'Right UP de 0° até 90°';
    G := TRUE;
    IF Right_Value > Right_MaxValue - Erro THEN
        G := FALSE;
        iRotP := 30;
        StateDesc := ' ';
    END_IF

30:
    // Right DOWN de 90° até 0°
    // deixar a Left em baixo
    // Considera-se estar Right_90 e Left_0
    StateDesc := 'Right DOWN de 90° até 0°';
    F := TRUE;
    B := TRUE;
    IF Right_Value < Right_MinValue + Erro - 100 AND Left_Value <
Left_MinValue + Erro - 100 THEN
        F := FALSE;
        B := FALSE;
        iRotP := 0;
        State := FALSE;
        StateDesc := ' ';
        LastOperation := 5;
    END_IF
END_CASE

Done := NOT State ;

```

Modo automático

Declaração de variáveis

VAR_INPUT

bReset : BOOL;
bEncostar : BOOL;
bRodar : BOOL;
bHome : BOOL;
BeamHorDim : REAL;
BeamVerDim : REAL;

END_VAR

VAR_OUTPUT

State : BOOL;
Done : BOOL;
Encostar_LED : BOOL;
Rodar_LED : BOOL;
Home_LED : BOOL;
bSusp : BOOL;
bPou : BOOL;
bPou_Short : BOOL;
bHor : BOOL;
PV_Hor : REAL;
Hor_Margin : REAL;
RotateDim : BOOL;

END_VAR

VAR

F1Desc : STRING;
F2Desc : STRING;

LastOperation : INT; // 1-Encostar , 2-Rodar , 3-Home
bEncostar_R_TRIG : R_Trig;
bRodar_R_TRIG : R_Trig;
bHome_R_Trig : R_Trig;

```

// Instanciação do controlo por ações - Macros
fbFlipper1_Act : FB_FLIPPER_ACT;
fbFlipper2_Act : FB_FLIPPER_ACT;

Timer : TON;
Time_Arrastamento : TIME := T#1.5S;

iEncostar : INT;
iRodar : INT;
iHome : INT;

HorizontalValue_mm_Mem1 : REAL;
HorizontalValue_mm_Mem2 : REAL;
END_VAR

```

Programa

```

IF bReset THEN
    iEncostar := 0;
    iRodar := 0;
    iHome := 0;
    State := FALSE;
    Encostar_LED := Rodar_LED := Home_LED := FALSE;
    F1Desc := "";
    F2Desc := "";
    Timer (IN := FALSE);
    LastOperation := 0;
END_IF

fbFlipper1_Act ( bReset := bReset , bSusp := FALSE , bPou := FALSE , bRotP :=
FALSE , bRotN := FALSE , bPou_Short := FALSE , bHor := FALSE , Erro :=
MAIN_PRG.Erro );
fbFlipper2_Act ( bReset := bReset , bSusp := FALSE , bPou := FALSE , bRotP :=
FALSE , bRotN := FALSE , bPou_Short := FALSE , bHor := FALSE , Erro :=
MAIN_PRG.Erro );

bEncostar_R_TRIG ( CLK := bEncostar );
bRodar_R_TRIG ( CLK := bRodar );
bHome_R_Trig (CLK := bHome );

CASE iHome OF

```

```

0:
    IF bHome_R_Trig.Q AND LastOperation <> 3 THEN
        State := TRUE; // manda ligar as bombas
        Home_LED := TRUE;
        iHome := 10;
        F1Desc := "";
        F2Desc := "";
        Timer (IN := FALSE);
    END_IF

10: // Pousa e fica com a pá esquerda a 90°
    fbFlipper1_Act ( bPou_Short := TRUE );
    fbFlipper2_Act ( bPou_Short := TRUE );
    Timer (IN := TRUE , PT := T#15S);
    IF (fbFlipper1_Act.State AND fbFlipper2_Act.State) THEN
        iHome := 11;
        fbFlipper1_Act ( bPou_Short := FALSE );
        fbFlipper2_Act ( bPou_Short := FALSE );
    END_IF

11:
    IF (fbFlipper1_Act.Done AND fbFlipper2_Act.Done) THEN
        Timer (IN := FALSE );
        iHome := 20;
        F1Desc := 'Pá Direita OK';
        F2Desc := 'Pá Direita OK';
    END_IF
    IF Timer.Q THEN
        Timer (IN := FALSE );
        IF NOT fbFlipper1_Act.Done THEN
            F1Desc := 'Problema na Pá Direita';
        ELSE
            F1Desc := 'Pá Direita OK';
        END_IF
        IF NOT fbFlipper2_Act.Done THEN
            F2Desc := 'Problema na Pá Direita';
        ELSE
            F2Desc := 'Pá Direita OK';
        END_IF
        iHome := 0;
        Home_LED := FALSE;
        State := FALSE;
        LastOperation := 3;
    END_IF

20: // Baixa as pás completamente
    fbFlipper1_Act ( bPou := TRUE );
    fbFlipper2_Act ( bPou := TRUE );
    Timer (IN := TRUE , PT := T#15S);
    IF (fbFlipper1_Act.State AND fbFlipper2_Act.State) THEN
        iHome := 21;
        fbFlipper1_Act ( bPou := FALSE );
        fbFlipper2_Act ( bPou := FALSE );

```

```

END_IF
21: //
IF (fbFlipper1_Act.Done AND fbFlipper2_Act.Done) THEN
    iHome := 30;
    F1Desc := 'Pá Esquerda OK';
    F2Desc := 'Pá Esquerda OK';
END_IF
IF Timer.Q THEN
    Timer (IN := FALSE );
    IF NOT fbFlipper1_Act.Done THEN
        F1Desc := 'Problema na Pá Esquerda';
    ELSE
        F1Desc := 'Pá Esquerda OK';
    END_IF
    IF NOT fbFlipper2_Act.Done THEN
        F2Desc := 'Problema na Pá Esquerda';
    ELSE
        F2Desc := 'Pá Esquerda OK';
    END_IF
    iHome := 0;
    Home_LED := FALSE;
    State := FALSE;
    LastOperation := 3;
END_IF

30:
fbFlipper1_Act ( bHor := TRUE , Hor_PV := 800, Hor_Margin := 0);
fbFlipper2_Act ( bHor := TRUE , Hor_PV := 800, Hor_Margin := 0);
Timer (IN := TRUE , PT := T#40S);
    IF (fbFlipper1_Act.State AND fbFlipper2_Act.State) THEN
        iHome := 31;
        fbFlipper1_Act ( bHor := FALSE );
        fbFlipper2_Act ( bHor := FALSE );
    END_IF

31:
IF (fbFlipper1_Act.Done AND fbFlipper2_Act.Done) THEN
    iHome := 0;
    Home_LED := FALSE;
    State := FALSE;
    LastOperation := 3;
    F1Desc := 'OK';
    F2Desc := 'OK';
END_IF
IF Timer.Q THEN
    Timer (IN := FALSE );
    IF NOT fbFlipper1_Act.Done THEN
        F1Desc := 'Problema no Pos. Horizontal';
    ELSE
        F1Desc := 'OK';
    END_IF
    IF NOT fbFlipper2_Act.Done THEN
        F2Desc := 'Problema no Pos. Horizontal';

```

```

ELSE
    F2Desc := 'OK';
END_IF
iHome := 0;
Home_LED := FALSE;
State := FALSE;
LastOperation := 3;
END_IF
END_CASE

// Operação de encostar a viga aos batentes
// Considera-se que a viga é pousada por cima da pá da direita
CASE iEncostar OF
    0: // Considera-se a viga colocada por cima da pá Direita
        IF bEncostar_R_Trig.Q THEN
            State := TRUE; // manda ligar as bombas
            Encostar_LED := TRUE;
            iEncostar := 10;
            Timer (IN := FALSE);
        END_IF

    10: // Suspenser
        fbFlipper1_Act ( bReset := FALSE , bSusp := TRUE );
        fbFlipper2_Act ( bReset := FALSE , bSusp := TRUE );
        IF (fbFlipper1_Act.State AND fbFlipper2_Act.State) THEN
            iEncostar := 11;
            fbFlipper1_Act ( bReset := FALSE , bSusp := FALSE );
            fbFlipper2_Act ( bReset := FALSE , bSusp := FALSE );
        END_IF

    11:
        IF (fbFlipper1_Act.Done AND fbFlipper2_Act.Done) THEN
            iEncostar := 20;
        END_IF

    20: // Mover até proximo do batente
        fbFlipper1_Act ( bHor := TRUE , Hor_PV := BeamHorDim, Hor_Margin :=
20); //Margem para arrastar
        fbFlipper2_Act ( bHor := TRUE , Hor_PV := BeamHorDim, Hor_Margin :=
20); //Margem para arrastar

        IF (fbFlipper1_Act.State AND fbFlipper2_Act.State) THEN
            iEncostar := 21;
            fbFlipper1_Act ( bHor := FALSE );
            fbFlipper2_Act ( bHor := FALSE );
        END_IF

    21:
        IF (fbFlipper1_Act.Done AND fbFlipper2_Act.Done) THEN
            iEncostar := 30;
        END_IF

    30: // Pousa e fica com a pá esquerda a 90°

```

```

fbFlipper1_Act ( bPou_Short := TRUE );
fbFlipper2_Act ( bPou_Short := TRUE );
    IF (fbFlipper1_Act.State AND fbFlipper2_Act.State) THEN
        iEncostar := 31;
        fbFlipper1_Act ( bPou_Short := FALSE );
        fbFlipper2_Act ( bPou_Short := FALSE );
    END_IF
31:
    IF (fbFlipper1_Act.Done AND fbFlipper2_Act.Done) THEN
        iEncostar := 40;
    END_IF

40:    // Encosta a viga aos batentes
        fbFlipper1_Act ( bHor := TRUE , Hor_PV := BeamHorDim, Hor_Margin := -
5); //Margem para arrastar
        fbFlipper2_Act ( bHor := TRUE , Hor_PV := BeamHorDim, Hor_Margin := -
5); //Margem para arrastar
            IF (fbFlipper1_Act.State AND fbFlipper2_Act.State) THEN
                iEncostar := 41;
                fbFlipper1_Act ( bHor := FALSE );
                fbFlipper2_Act ( bHor := FALSE );
            END_IF
41:
        Timer ( IN := TRUE , PT := Time_Arrastamento );
        IF Timer.Q THEN
            fbFlipper1_Act ( bReset := TRUE );
            fbFlipper2_Act ( bReset := TRUE );
            Timer ( IN := FALSE );
            iEncostar := 50;
        END_IF

50:    // Margem de segurança
        fbFlipper1_Act ( bHor := TRUE , Hor_PV := BeamHorDim, Hor_Margin :=
5);
        fbFlipper2_Act ( bHor := TRUE , Hor_PV := BeamHorDim, Hor_Margin :=
5);
            IF (fbFlipper1_Act.State AND fbFlipper2_Act.State) THEN
                iEncostar := 51;
                fbFlipper1_Act ( bHor := FALSE );
                fbFlipper2_Act ( bHor := FALSE );
            END_IF
51:
        IF (fbFlipper1_Act.Done AND fbFlipper2_Act.Done) THEN
            iEncostar := 60;
        END_IF

60: // Baixa as pás completamente
        fbFlipper1_Act ( bPou := TRUE );
        fbFlipper2_Act ( bPou := TRUE );
            IF (fbFlipper1_Act.State AND fbFlipper2_Act.State) THEN
                iEncostar := 61;

```

```

        fbFlipper1_Act ( bPou := FALSE );
        fbFlipper2_Act ( bPou := FALSE );
    END_IF
61: //
    IF (fbFlipper1_Act.Done AND fbFlipper2_Act.Done) THEN
        iEncostar := 70;
    END_IF

70:    //A pá tem de ficar dentro da estrutura
    IF fbFlipper1_Act.Horizontal_Value_mm < 400 THEN
        fbFlipper1_Act ( bHor := TRUE , Hor_PV := 400, Hor_Margin := 20);
        fbFlipper2_Act ( bHor := TRUE , Hor_PV := 400, Hor_Margin := 20);
        IF (fbFlipper1_Act.State AND fbFlipper2_Act.State) THEN
            iEncostar := 71;
            fbFlipper1_Act ( bHor := FALSE );
            fbFlipper2_Act ( bHor := FALSE );
        END_IF
    ELSE;
        iEncostar := 0;
        Encostar_LED := FALSE;
        State := FALSE;
        LastOperation := 1;
    END_IF

71:    IF (fbFlipper1_Act.Done AND fbFlipper2_Act.Done) THEN
        iEncostar := 0;
        Encostar_LED := FALSE;
        State := FALSE;
        LastOperation := 1;
    END_IF
END_CASE

CASE iRodar OF
    0:
        IF bRodar_R_Trig.Q THEN
            Rodar_LED := TRUE;
            State := TRUE; // manda ligar as bombas
            RotateDim := FALSE;
            IF LastOperation = 2 THEN //Rodar diretamente , sem ir para perto da
viga (já está lá à beira)
                iRodar := 12;
            ELSE
                iRodar := 10;
            END_IF
        END_IF

    10: // Vai para perto da viga
        fbFlipper1_Act ( bHor := TRUE , Hor_PV := BeamHorDim, Hor_Margin :=
5);
        fbFlipper2_Act ( bHor := TRUE , Hor_PV := BeamHorDim, Hor_Margin :=
5);
        IF (fbFlipper1_Act.State AND fbFlipper2_Act.State) THEN

```



```

        iRodar := 11;
        fbFlipper1_Act ( bHor := FALSE );
        fbFlipper2_Act ( bHor := FALSE );
    END_IF
11:
    IF (fbFlipper1_Act.Done AND fbFlipper2_Act.Done) THEN
        iRodar := 12;
    END_IF

12: // Suspende
    fbFlipper1_Act ( bReset := FALSE , bSusp := TRUE );
    fbFlipper2_Act ( bReset := FALSE , bSusp := TRUE );
    IF (fbFlipper1_Act.State AND fbFlipper2_Act.State) THEN
        iRodar := 13;
        fbFlipper1_Act ( bReset := FALSE , bSusp := FALSE );
        fbFlipper2_Act ( bReset := FALSE , bSusp := FALSE );
    END_IF
13:
    IF (fbFlipper1_Act.Done AND fbFlipper2_Act.Done) THEN
        iRodar := 15;
    END_IF

15:
    fbFlipper1_Act ( bHor := TRUE , Hor_PV := 80, Hor_Margin := 0);
    fbFlipper2_Act ( bHor := TRUE , Hor_PV := 80, Hor_Margin := 0);
    IF (fbFlipper1_Act.State AND fbFlipper2_Act.State) THEN
        iRodar := 16;
        fbFlipper1_Act ( bHor := FALSE );
        fbFlipper2_Act ( bHor := FALSE );
    END_IF
16:
    IF (fbFlipper1_Act.Done AND fbFlipper2_Act.Done) THEN
        iRodar := 20;
    END_IF

20: // Roda a Viga
    fbFlipper1_Act ( bRotP_Susp := TRUE );
    fbFlipper2_Act ( bRotP_Susp := TRUE );
    IF (fbFlipper1_Act.State AND fbFlipper2_Act.State) THEN
        iRodar := 21;
        fbFlipper1_Act ( bRotP_Susp := FALSE );
        fbFlipper2_Act ( bRotP_Susp := FALSE );
        HorizontalValue_mm_Mem1 :=
fbFlipper1_Act.Horizontal_Value_mm;
        HorizontalValue_mm_Mem2 :=
fbFlipper2_Act.Horizontal_Value_mm;
    END_IF
21:
    IF (fbFlipper1_Act.Done AND fbFlipper2_Act.Done) THEN

```

```

        iRodar := 30;
    END_IF

    30:    // Mov horizonatal para posição de suspender
           fbFlipper1_Act ( bHor := TRUE , Hor_PV := 100 +
HorizontalValue_mm_Mem1 + BeamVerDim , Hor_Margin := 5);
           fbFlipper2_Act ( bHor := TRUE , Hor_PV := 100 +
HorizontalValue_mm_Mem2 + BeamVerDim , Hor_Margin := 5);
           IF (fbFlipper1_Act.State AND fbFlipper2_Act.State) THEN
               iRodar := 31;
               fbFlipper1_Act ( bHor := FALSE );
               fbFlipper2_Act ( bHor := FALSE );
           END_IF
    31:    IF (fbFlipper1_Act.Done AND fbFlipper2_Act.Done) THEN
               iRodar := 0;
               Rodar_LED := FALSE;
               State := FALSE;
               LastOperation := 2;
               RotateDim := TRUE;
           END_IF
END_CASE

Done := NOT State;

```